LNCS 10858

Roger Nkambou
Roger Azevedo
Julita Vassileva (Eds.)

Intelligent Tutoring Systems

14th International Conference, ITS 2018
Montreal, QC, Canada, June 11–15, 2018
Proceedings

Springer

Lecture Notes in Computer Science 10858

More information about this series at http://www.springer.com/series/7408

Roger Nkambou · Roger Azevedo
Julita Vassileva (Eds.)

Intelligent Tutoring Systems

14th International Conference, ITS 2018
Montreal, QC, Canada, June 11–15, 2018
Proceedings

Springer

Editors

Roger Nkambou
Université du Québec
Montreal, QC
Canada

Julita Vassileva
University of Saskatchewan
Saskatoon, SK
Canada

Roger Azevedo
NCSU
Raleigh, NC
USA

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Computer Science
ISBN 978-3-319-91463-3 ISBN 978-3-319-91464-0 (eBook)
https://doi.org/10.1007/978-3-319-91464-0

Library of Congress Control Number: 2018942342

LNCS Sublibrary: SL2 – Programming and Software Engineering

Printed on acid-free paper

This Springer imprint is published by the registered company Springer International Publishing AG
part of Springer Nature
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

The 14th International Conference on Intelligent Tutoring Systems (ITS 2018) was held in the birth city of the ITS conferences, Montreal, Canada, during June 11–15, 2018.

The theme of ITS 2018 was "A 30-Year Legacy of ITS Conferences" with an objective to celebrate the academic and research achievements as well as the ongoing scientific contributions and impact of the ITS conferences over a 30-year history in the field of intelligent systems in education and across other disciplines. The conference emphasized the use of advanced computer technologies and interdisciplinary research for enabling, supporting, and enhancing human learning. It promoted high-quality interdisciplinary research creating a forum for presenting and sharing challenges and novel advancements in artificial intelligence. It triggered an exchange of ideas in the field, reinforcing and expanding the international ITS network of researchers, academics, and market representatives. The call for scientific papers solicited work presenting substantive new research results in using advanced computer technologies and interdisciplinary research for enabling, supporting, and enhancing human learning. A Posters Track was also organized, which provided an interactive forum for authors to present research prototypes to conference participants, as well as work in progress.

The international Program Committee consisted of 93 leading members (43 senior and 50 regular) of the intelligent Tutoring Systems community, assisted by 33 external reviewers. The conference (general) chair was Roger Nkambou from the Université du Québec à Montréal (UQAM), Canada; the Program Committee chairs were Roger Azevedo from North Carolina State University, USA, and Julita Vassileva from the University of Saskatchewan, Canada.

Research papers were reviewed by at least three reviewers (with the majority receiving four or more reviews) through a double-blind process. Only 26.5% of papers submitted as full paper were accepted; 21 were accepted as short papers with six pages published in the proceedings. We also accepted 29 posters presentations; some of them were directly submitted to the poster track chaired by Bob Hausman (Carnegie Learning) and Tanner Jackson (Educational, Testing Service, USA). We believe that the selected full papers describe some very significant research and the short papers some very interesting new ideas, while the posters present research in progress that deserves close attention.

A separate Doctoral Consortium (DC) provided a forum in which PhD students could present and discuss their work during its early stages, meet peers with shared interests, and receive feedback from senior members of the field as mentors. The DC chairs were Maiga Chang from Athabasca University, Canada, and Éric Beaudry from UQAM, Canada. The DC Committee accepted six papers; each of them received at least five reviews with many (50%) receiving seven reviews. The management of the review process and the preparation of the proceedings was handled through EasyChair.

Additionally, the ITS 2018 program included the following workshops and tutorials selected by the workshop and tutorial chairs, Nathalie Guin from the Université de Lyon 1, France, and Amruth Kumar from Ramapo College of New Jersey, USA:

– W1: C&C-ITS - Context and Culture in Intelligent Tutoring Systems by Valéry Psyche, Isabelle Savard, Riichiro Mizoguchi and Jacqueline Bourdeau
– W2: Learning Analytics: Building Bridges Between the Education and the Computing Communities by Sébastien Beland, Michel Desmarais and Nathalie Loye
– W3: Exploring Opportunities for Caring Assessments by Diego Zapata-Rivera and Julita Vassileva
– W6 (FD): Optimizing Human Learning: Workshop eliciting Adaptive Sequences for Learning (WeASeL), Jill-Jênn Vie and Fabrice Popineau
– T1: Automating Educational Research Through Learning Analytics: Data Balancing and Matching Techniques by David Boulanger, Vivekanandan Kumar and Shawn Fraser
– T2: Authoring, Deploying and Data Analysis of Conversational Intelligent Tutoring Systems by Xiangen Hu, Zhiqiang Cai, Arthur Graesser and Keith Shubeck.

The ITS 2018 industry track included the following workshops selected by the industry track chair, Robert Sottilare from the US Army Research Laboratory, USA:

– W7: ITS GIFT Workshop
– W8: ITS Standards for Adaptive Instructional Systems Workshop

Finally, we had three outstanding invited speakers in the plenary sessions: Yoshua Bengio (University of Montreal, Canada), a renowned figure in the field of deep learning, Vania Dimitrova (University of Leeds, UK), and Sidney D'Mello (University of Colorado Boulder, USA), both leaders in different specialized areas of the ITS field.

In addition to the contributors mentioned above, we would like to thank all the authors, the various conference chairs, the members of the Program Committees of all tracks, the external reviewers, the Steering Committee members and in particular its chair, Claude Frasson. We would also like to acknowledge the Institute of Intelligent Systems (IIS), which acted as the conference organizer (particularly Kitty Panourgia and her excellent team for the permanent follow up of the organization), and the Université du Québec à Montréal (UQAM), the hosting institution. Last but not least, we express our gratitude to the conference sponsors, in particular the National Science Foundation (NSF), Springer, and Tourisme Montreal for their financial support.

June 2018

Roger Nkambou
Roger Azevedo
Julita Vassileva

Organization

Conference Committees

Conference (General) Chair

Roger Nkambou	Université du Québec à Montréal, Canada

Program Chairs

Roger Azevedo	North Carolina State University, USA
Julita Vassileva	University of Saskatchewan, Canada

Organization Chair

Kitty Panourgia	Neoanalysis, Greece

Workshop and Tutorial Chairs

Nathalie Guin	Université Lyon 1, France
Amruth Kumar	Ramapo College of New Jersey, USA

Posters and Demos Chairs

G. Tanner Jackson	Educational Testing Service, USA
Bob Hausmann	Carnegie Mellon, USA

Doctoral Consortium Chairs

Maiga Chang	Athabasca University, Canada
Éric Beaudry	Université du Québec à Montréal, Canada

Industry Track Chair

Robert Sottilare	US Army Research Laboratory, USA

Sponsorship and Exhibition Chairs

Emmanuel Blanchard	IDU Interactive Inc., Canada
Sophie Callies	Ellicom, Canada
Valéry Psyché	Téluq, Canada

Student Volunteer Chairs

Ramla Gali	Université du Québec à Montréal, Canada
Ange Tato	Université du Québec à Montréal, Canada

The conference is held under the auspices of the Institute of Intelligent Systems.

iiS

INSTITUTE OF
INTELLIGENT SYSTEMS

Program Committee

Chairs

Roger Azevedo	North Carolina State University, USA
Julita Vassileva	University of Saskatchewan, Canada

Senior Program Committee

Esma Aimeur	University of Montreal, Canada
Ivon Arroyo	Worcester Polytechnic Institute, USA
Kevin Ashley	University of Pittsburgh, USA
Ryan Baker	Teachers College, Columbia University, USA
Jacqueline Bourdeau	TELU-UQAM, Canada
Bert Bredeweg	University of Amsterdam, The Netherlands
Paul Brna	University of Leeds, UK
Stefano A. Cerri	LIRMM: University of Montpellier and CNRS, France
Maiga Chang	Athabasca University, Canada
Michaela Cocea	University of Porthsmouth, UK
Cristina Conati	University of British Columbia, Canada
Albert Corbett	Carnegie Mellon University, USA
Michel Desmarais	Ecole Polytechnique de Montreal, Canada
Vania Dimitrova	University of Leeds, UK
Benedict Du Boulay	University of Sussex, UK
Peter Dolog	Aalborg University, Denmark
Claude Frasson	University of Montreal, Canada
Gilles Gauthier	Université du Québec à Montréal, Canada
Art Graesser	University of Memphis, USA
Sabine Graf	Athabasca University, Canada
Jim Greer	University of Saskatchewan, Canada
Tsukasa Hirashima	Hiroshima University, Japan

W. Lewis Johnson	Alelo Inc., USA
Kinshuk Kinshuk	University of North Texas, USA
Vive Kumar	Athabasca University, Canada
Jean-Marc Labat	Université Paris 6, France
Susanne Lajoie	McGill University, Canada
James Lester	North Carolina State University, USA
Vanda Luengo	Université Pierre et Marie Curie, France
Gordon McCalla	University of Saskatchewan, Canada
Alessandro Micarelli	University of Rome 3, Italy
Riichiro Mizoguchi	Japan Advanced Institute of Science and Technology, Japan
Roger Nkambou	Université du Québec à Montréal, Canada
Anna Paiva	Technical University of Lisbon, Portugal
Niels Pinkwart	Humboldt Universität zu Berlin, Germany
Elvira Popescu	University of Craiova, Romania
Carolyn Rose	Carnegie Mellon University, USA
Demetrios Sampson	University of Piraeus, Greece
Stefan Trausan-Matu	Politehnica University of Bucharest, Romania
Kurt VanLehn	Arizona State University, USA
Julita Vassileva	University of Saskatchewan, Canada
Barbara Wasson	University of Bergen, Norway
Beverly Park Woolf	University of Massachusetts, USA

Program Committee

Mohammed Abdel Razek	King Abdulaziz University, Saudi Arabia
Fabio Akhras	Renato Archer Center of Information Technology, Brazil
Galia Angelova	Bulgarian Academy of Sciences, Bulgaria
Maria Lucia Barron-Estrada	Instituto Tecnologico de Culiacan, Mexico
Maria Bielikova	Slovak University of Technology in Bratislava, Slovakia
Emmanuel Blanchard	IDÛ Interactive Inc., Canada
Stephen B. Blessing	University of Tampa, USA
Mary Jean Blink	TutorGen, Inc., USA
François Bouchet	Sorbonne University, France
Ted Carmichael	University of North Carolina at Charlotte, USA
Chih-Kai Chang	National University of Tainan, Taiwan
Maher Chaouachi	McGill University, Canada
Min Chi	North Carolina State University, USA
Chih-Yueh Chou	Yuan Ze University, Taiwan
Mark Core	University of Southern California, USA
Evandro Costa	Federal University of Alagoas, Brazil
Alexandra Cristea	University of Warwick, UK
Diego Demerval	Federal University of Alagoas, Brazil
Cyrille Desmoulins	Université Joseph Fourier, France

Philippe Dessus	LSE, Grenoble, France
Darina Dicheva	Winston Salem State University, USA
Sidney D'Mello	University of Colorado Boulder, USA
Stephen Fancsali	Carnegie Learning, Inc., USA
Robert Farrell	IBM T.J. Watson Research Center, USA
Mark Floryan	University of Virginia, USA
Davide Fossati	Carnegie Mellon University, Qatar
Ashok Goel	Georgia Institute of Technology, USA
Jason Harley	University of Alberta, Canada
Yusuke Hayashi	Hiroshima University, Japan
Cecily Heiner	Southern Utah University, USA
Seiji Isotani	University of Sao Paulo, Brazil
Patricia Jaques	UNISINOS, Brazil
Imène Jraidi	University of Montreal, Canada
Akihiro Kashihara	University of Electro-Communications, Japan
Nguyen-Thinh Le	Humboldt Universität zu Berlin, Germany
H. Chad Lane	University of Illinois, Urbana-Champaign, USA
Elise Lavoué	University of Lyon, France
Carla Limongelli	University of Rome 3, Italy
Fuhua Lin	Athabasca University, Canada
Tatsunori Matsui	Waseda University, Japan
Riccardo Mazza	University of Lugano/University of Applied Sciences of Southern Switzerland
Tassos Mikopoulos	University of Ioannina, Greece
Kazuhisa Miwa	Nagoya University, Japan
Luc Paquette	University of Illinois, Urbana-Champaign, USA
Zach Pardos	UC Berkeley, USA
Alexandra Poulovassilis	Birkbeck University of London, UK
Jonathan Rowe	North Carolina State University, USA
Olga C. Santos	aDeNu Research Group, UNED, Spain
Michelle Taub	North Carolina State University, USA
Ramon Zatarain Cabada	Instituto Tecnologico de Culiacan, Mexico

Organizing Committee

Chair

Kitty Panourgia	Coordination

Members

Mara Gassel	Conference Publicity/Website Management/Registration
Alexia Kakourou	Coordination on Site
Katerina Milathianaki	Preparation/Administration
Dimitris Sevastakis	IT Support
Isaak Tselepis	Website Architect

Steering Committee

Chair

Claude Frasson	University of Montreal, Canada

Members

Stefano A. Cerri	LIRMM: University of Montpellier and CNRS, France
Isabel Fernandez-Castro	University of the Basque Country, Spain
Gilles Gauthier	Université du Québec à Montréal, Canada
Guy Gouardères	University of Pau, France
Tsukasa Hirashima	University of Hiroshima, Japan
Marc Kaltenbach	Bishop’s University, Canada
Alan Lesgold	University of Pittsburgh, USA
James Lester	North Carolina State University, USA
Alessandro Micarelli	University of Rome 3, Italy
Roger Nkambou	Université du Québec à Montréal, Canada
Giorgos Papadourakis	Technological Educational Institute, Crete, Greece
Fabio Paragua	Federal University of Alagoas, Brazil
Elliot Soloway	University of Michigan, USA
Daniel Suthers	University of Hawaii, USA
Stefan Trausen-Matu	University Politehnica of Bucharest, Romania
Beverly Woolf	University of Massachusetts, USA

Advisory Committee

Luigia Carlucci Aiello	University of Rome, Italy
Maria Grigoriadou	University of Athens, Greece
Judy Kay	University of Sydney, Australia
Demetrios G. Sampson	Curtin University, Australia

Poster and Interactive Event Committee

Chairs

Bob Hausmann	Carnegie Mellon, USA
G. Tanner Jackson	Educational Testing Service, USA

Members

Ryan Baker	University of Pennsylvania, USA
Scotty Craig	Arizona State University, USA
Reva Freedman	Northern Illinois University, USA
Chas Murray	Carnegie Learning, Inc., USA
Amy Johnson	Arizona State University, USA
Blair Lehman	Educational Testing Service, USA
Rod Roscoe	Arizona State University, USA

Doctoral Consortium Committee

Chairs

Éric Beaudry	Université du Québec à Montréal, Canada
Maiga Chang	Athabasca University, Canada

Members

Ben Chang	National Central University, Taiwan
Nian-Shing Chen	National Sun Yat-sen University, Taiwan
Xiaoqing Gu	East China Normal University, China
Gwo-Jen Hwang	National Taiwan University of Science and Technology, Taiwan
Kinshuk Kinshuk	College of Information, University of North Texas, USA
Vive Kumar	Athabasca University, Canada
Rita Kuo	New Mexico Institute of Mining and Technology, USA
Kuo-Chen Li	CYCU
Oscar Lin	Athabasca University, Canada
Wolfgang Mueller	University of Education Weingarten, Germany
Kuo-Liang Ou	National Hsin-Chu University of Education, Taiwan
Dongming Qian	East China Normal University, China
Dunwei Wen	Athabasca University, USA

Additional Reviewers

Sungeun An
Jason Bernard
Joana Campos
Christopher Cassion
Joao Dias
Bobbie Eicher
Stephanie Frost
Marissa Gonzales
Avery Harrison
Yugo Hayashi
OluwabUKola Ishola
Ondrej Kaššák
Sébastien Lallé
David Edgar Lelei
Chen Lin
Yang Liu
Alexandra Luccioni
Leonardo Marques
Samuel Mascarenhas
Miki Matsumuro
Nicholas Mudrick
Ivelina Nikolova
Sydni Peterson
Diogo Rato
Tamra Ross
Hitomi Saito
Salvatore Vanini
Naomi Wixon
Mohammad Belghis-Zadeh
Yuan Zhang
Guojing Zhou

Conference Sponsors

The National Science Foundation awarded a grant to support travel expenses for American students from universities in the USA who participated in the Doctoral Consortium (DC) at ITS 2018.

Springer awarded a prize of 1,000 Euro for the best ITS 2018 full paper.

Tourisme Montreal sponsored the 14th International Conference on Intelligent Tutoring Systems providing 30 CAD per participant who came to attend the ITS 2018 Conference from any region outside the Province of Quebec.

Invited Talks

Deep Learning and Cognition

Yoshua Bengio

Department of Computer Science and Operational Research,
University of Montreal, Montreal, Canada
yoshua.umontreal@gmail.com

Abstract. Neural networks and deep learning have been inspired by brains, neuroscience and cognition, from the very beginning, starting with distributed representations, neural computation, and the hierarchy of learned features. More recently, it has been for example with the use of rectifying non-linearities (ReLU) – which enables training deeper networks – as well as the use of soft content-based attention – which allow neural nets to go beyond vectors and to process a variety of data structures and led to a breakthrough in machine translation. Ongoing research is now suggesting that brains may use a process similar to backpropagation for estimating gradients and new inspiration from cognition suggests how to learn deep representations which disentangle the underlying factors of variation, by allowing agents to intervene and explore in their environment.

Speaker Bio. Yoshua Bengio (computer science, 1991, McGill U; post-docs at MIT and Bell Labs, computer science professor at U. Montréal since 1993): he authored three books, over 300 publications (h-index over 100), mostly in deep learning, holds a Canada Research Chair in Statistical Learning Algorithms, is Officer of the Order of Canada, recipient of the Marie-Victorin Quebec Prize 2017, he is a CIFAR Senior Fellow and co-directs its Learning in Machines and Brains program. He heads the Montreal Institute for Learning Algorithms (MILA), currently the largest academic research group on deep learning. He is on the NIPS foundation board (previously program chair and general chair) and co-created the ICLR conference (specialized in deep learning). He pioneered deep learning and his goal is to uncover the principles giving rise to intelligence through learning, as well as contribute to the development of AI for the benefit of all. Yoshua Bengio is considered as one of the three fathers of an advanced subset of AI and machine learning called deep learning and has helped Montreal to become the Silicon Valley of AI!

From Intelligent Tutors to Intelligent Mentors: Looking Back into the Future

Vania Dimitrova

University of Leeds, UK
V.G.Dimitrova@leeds.ac.uk

Abstract. 20 years ago in his invited talk at ITS98, John Self outlined the defining characteristics of intelligent tutoring systems, pointing that these systems adapt to the needs of learners and provide some degree of computational precision. I will revisit these characteristics in the light of 21st Century education challenges, pointing that the time is ripe for the emergence of a new breed of intelligent tutors that provide mentor-like features. Mentoring is seen as a highly effective method to support the development of transferable skills, to increase motivation and confidence, and to develop self-regulation and self-determination. However, mentoring does not scale and can be costly - while 'everyone needs a mentor' not everyone can have a mentor. With the abundance of digital content and digital traces that capture our behaviour in the physical world, there is an opportunity to develop intelligent mentors. They would require multi-faceted 'learner sensing' mechanisms to get sufficient understanding of the learner's engagement and motivation by analysing the various digital traces left by the learner or by other learners. Furthermore, intelligent mentors will embed strategies for promoting reflection and self-awareness through 'personalised nudges'. I will illustrate this vision with two ongoing projects: the Active Video Watching project that develops interactive means for engaging with videos for transferable skills learning, and the myPAL project that provides a personalised adaptive learning companion for self-regulated learning.

Speaker Bio. Vania Dimitrova (PhD in AI in Education, Leeds) is Associate Professor in Intelligent Systems in the School of Computing, University of Leeds, Co-director of the Leeds University Research Centre in Digital Learning, and Director of Technology-enhanced learning strategy at the Leeds Institute of Medical Education. She leads a research activity on AI for augmenting human intelligence. This develops methods for knowledge capture, ontological modelling and reasoning, user/group modelling, and user-adaptive interactive systems. She is involved in several multi-disciplinary projects on (a) knowledge-enriched intelligent decision systems and (b) user behaviour modelling for intelligent support for self-regulation and soft skills learning. Her work is funded by a range of sources, e.g. EU, UK research councils, the UK technology strategy board. She coordinated ImREAL (EU) that developed culturally-enhanced personalised simulators for learning, and led personalisation for decision making in Dicode (EU). She is associate editor of the International Journal of AI in Education (IJAIED), member of the editorial board of the personalisation journal (UMUAI), and was associate editor of IEEE Transactions on Learning Technologies (IEEE TLT). She is a member of the executive Committee of the International AI in Education society; and regularly acts as EU projects reviewer.

Distributed Cognition in Multimodal Collaborative Learning Environments

Sidney K. D'Mello

Department of Computer Science and Institute of Intelligent Systems,
University of Colorado Boulder, Boulder, USA
sidney.dmello@colorado.edu

Abstract. Distributed cognition (DCog) pertains to cognition that extends beyond the individual to collections of interacting individuals and their environment. It is distinct from traditional cognition in that the unit of analysis is a system not an individual. It involves socio-cognitive-affective processes that are multimodal, interact over multiple spatial and temporal scales, and are embedded in a constantly-changing environment. I will discuss projects aimed at uncovering basic principles of distributed cognition in multimodal collaborative learning environments with an eye towards incorporating insights into next-generation learning technologies that aim to improve collaborative processes and outcomes.

Speaker Bio. Sidney D'Mello (PhD in Computer Science) is an Associate Professor in the Institute of Cognitive Science and Department of Computer Science at the University of Colorado Boulder. He is interested in the dynamic interplay between cognition and emotion while individuals and groups engage in complex real-world tasks. He applies insights gleaned from this basic research program to develop intelligent technologies that help people achieve to their fullest potential by coordinating what they think and feel with what they know and do. D'Mello has co-edited six books and published over 220 journal papers, book chapters, and conference proceedings (13 of these have received awards). His work has been funded by numerous grants and he serves(d) as associate editor for four journals, on the editorial boards for six others, and has played leadership roles in several professional organizations. https://www.colorado.edu/ics/sidney-dmello.

Contents

Full Papers

Programming Intelligent Embodied Pedagogical Agents to Teach the Beginnings of Industrial Revolution 3
Ivan Luis Feix Baierle and João Carlos Gluz

Adaptive Clustering of Codes for Assessment in Introductory Programming Courses 13
Alexandre de A. Barbosa, Evandro de B. Costa, and Patrick H. Brito

Recommendation in Collaborative E-Learning by Using Linked Open Data and Ant Colony Optimization 23
Samia Beldjoudi, Hassina Seridi, and Nour El Islem Karabadji

Evaluating Adaptive Pedagogical Agents' Prompting Strategies Effect on Students' Emotions 33
François Bouchet, Jason M. Harley, and Roger Azevedo

Investigating the Role of Goal Orientation: Metacognitive and Cognitive Strategy Use and Learning with Intelligent Tutoring Systems 44
Elizabeth B. Cloude, Michelle Taub, and Roger Azevedo

Game Scenes Evaluation and Player's Dominant Emotion Prediction 54
René Doumbouya, Mohamed S. Benlamine, Aude Dufresne, and Claude Frasson

Disrupting the Rote Learning Loop: CS Majors Iterating Over Learning Modules with an Adaptive Educational Hypermedia 66
Muhammad Mustafa Hassan and Adnan N. Qureshi

Gaze Feedback and Pedagogical Suggestions in Collaborative Learning: Investigation of Explanation Performance on Self's Concept in a Knowledge Integration Task 78
Yugo Hayashi

Motivational Assessment Tool (MAT): Enabling Personalized Learning to Enhance Motivation 88
Elizabeth Lameier, Lauren Reinerman-Jones, Gerald Matthews, Elizabeth Biddle, and Michael Boyce

The Impact of Multiple Real-Time Scaffolding Experiences on Science Inquiry Practices 99
Haiying Li, Janice Gobert, Rachel Dickler, and Raha Moussavi

The Allocation of Time Matters to Students' Performance in Clinical Reasoning 110
Shan Li, Juan Zheng, Eric Poitras, and Susanne Lajoie

Empirical Investigation of Cognitive Load Theory in Problem Solving Domain 120
Kazuhisa Miwa, Hitoshi Terai, and Kazuaki Kojima

Data-Driven Learner Profiling Based on Clustering Student Behaviors: Learning Consistency, Pace and Effort. 130
Shirin Mojarad, Alfred Essa, Shahin Mojarad, and Ryan S. Baker

Identifying How Metacognitive Judgments Influence Student Performance During Learning with MetaTutorIVH. 140
Nicholas V. Mudrick, Robert Sawyer, Megan J. Price, James Lester, Candice Roberts, and Roger Azevedo

Predicting Learners' Emotions in Mobile MOOC Learning via a Multimodal Intelligent Tutor. 150
Phuong Pham and Jingtao Wang

Toward Tutoring Systems Inspired by Applied Behavioral Analysis 160
Michela Ponticorvo, Angelo Rega, and Orazio Miglino

The Role of Negative Emotions and Emotion Regulation on Self-Regulated Learning with MetaTutor 170
Megan J. Price, Nicholas V. Mudrick, Michelle Taub, and Roger Azevedo

Analysis of Permanence Time in Emotional States: A Case Study Using Educational Software 180
Helena Reis, Danilo Alvares, Patricia Jaques, and Seiji Isotani

Scoring Summaries Using Recurrent Neural Networks. 191
Stefan Ruseti, Mihai Dascalu, Amy M. Johnson, Danielle S. McNamara, Renu Balyan, Kathryn S. McCarthy, and Stefan Trausan-Matu

Changes in Emotion and Their Relationship with Learning Gains in the Context of MetaTutor. 202
Jeanne Sinclair, Eunice Eunhee Jang, Roger Azevedo, Clarissa Lau, Michelle Taub, and Nicholas V. Mudrick

A Heuristic Approach for New-Item Cold Start Problem in Recommendation of Micro Open Education Resources 212
Geng Sun, Tingru Cui, Dongming Xu, Jun Shen, and Shiping Chen

How Do Different Levels of AU4 Impact Metacognitive Monitoring During Learning with Intelligent Tutoring Systems? 223
Michelle Taub, Roger Azevedo, and Nicholas V. Mudrick

How Are Students' Emotions Associated with the Accuracy of Their Note Taking and Summarizing During Learning with ITSs? 233
Michelle Taub, Nicholas V. Mudrick, Ramkumar Rajendran, Yi Dong, Gautam Biswas, and Roger Azevedo

Adaptive Feedback Based on Student Emotion in a System for Programming Practice 243
Thomas James Tiam-Lee and Kaoru Sumi

Learning by Explaining to a Digital Doppelganger 256
Ning Wang, Ari Shapiro, Andrew Feng, Cindy Zhuang, Chirag Merchant, David Schwartz, and Stephen L. Goldberg

Short Papers

Impact of Tutor Errors on Student Engagement in a Dialog Based Intelligent Tutoring System 267
Shazia Afzal, Vinay Shashidhar, Renuka Sindhgatta, and Bikram Sengupta

Enhancing the Clustering of Student Performance Using the Variation in Confidence 274
Ani Aghababyan, Nicholas Lewkow, and Ryan S. Baker

Intelligent Virtual Reality Tutoring System Supporting Open Educational Resource Access 280
Jae-wook Ahn, Ravi Tejwani, Sharad Sundararajan, Aldis Sipolins, Sean O'Hara, Anand Paul, Ravi Kokku, Jan Kjallstrom, Nam Hai Dang, and Yazhou Huang

Emotional State and Behavior Analysis in a Virtual Reality Environment: A Medical Application 287
Hamdi Ben Abdessalem, Marwa Boukadida, and Claude Frasson

Deep Learning in Automated Essay Scoring 294
David Boulanger and Vivekanandan Kumar

A Hybrid Architecture for Non-technical Skills Diagnosis 300
Yannick Bourrier, Francis Jambon, Catherine Garbay, and Vanda Luengo

A Novel Learning Early-Warning Model Based on Random Forest Algorithm . . . 306
Xiaoxiao Cheng, Zhengzhou Zhu, Xiao Liu, Xiaofang Yuan, Jiayu Guo, Qun Guo, Deqi Li, and Ruofei Zhu

Improving Inference of Learning Related Emotion by Combining Cognitive and Physical Information . . . 313
Ernani Gottardo and Andrey Ricardo Pimentel

Module Advisor: Guiding Students with Recommendations . . . 319
Nina Hagemann, Michael P. O'Mahony, and Barry Smyth

TUMA: Towards an Intelligent Tutoring System for Manual-Procedural Activities . . . 326
Zardosht Hodaie, Juan Haladjian, and Bernd Bruegge

SAT Reading Analysis Using Eye-Gaze Tracking Technology and Machine Learning . . . 332
Andrew Howe and Phong Nguyen

Determining What the Student Understands - Assessment in an Unscaffolded Environment . . . 339
C. W. Liew and H. Nguyen

Curriculum Pacing: A New Approach to Discover Instructional Practices in Classrooms . . . 345
Nirmal Patel, Aditya Sharma, Collin Sellman, and Derek Lomas

Towards Embedding a Tutoring Companion in the Eclipse Integrated Development Environment . . . 352
Manohara Rao Penumala and Javier Gonzalez-Sanchez

Semantic Collaboration Trajectories in Communities of Practice . . . 359
Matheus Pereira, Rosa Maria Vicari, and João Luis Tavares da Silva

Predictors and Outcomes of Gaming in an Intelligent Tutoring System . . . 366
Chad Peters, Ivon Arroyo, Winslow Burleson, Beverly Woolf, and Kasia Muldner

Classifying Interaction Behaviors of Students and Conversational Agents Through Dialog Analysis . . . 373
Michael Procter, Robert Heller, and Fuhua Lin

Extraction of Relevant Resources and Questions from DBpedia to Automatically Generate Quizzes on Specific Domains . . . 380
Oscar Rodríguez Rocha, Catherine Faron Zucker, and Alain Giboin

A Planning-Based Approach to Generating Tutorial Dialog for Teaching Surgical Decision Making 386
Narumol Vannaprathip, Peter Haddawy, Holger Schultheis, Siriwan Suebnukarn, Parichat Limsuvan, Atirach Intaraudom, Nattapon Aiemlaor, and Chontee Teemuenvai

Supporting Multiple Learning Experiences on a Childhood Vocabulary Tutoring Platform 392
Aditya Vempaty, Tamer Abuelsaad, Allison Allain, and Ravi Kokku

Exploring Students' Behaviors in Editing Learning Environment. 399
Xuebai Zhang, Xiaolong Liu, Shyan-Ming Yuan, Chia-Chen Fan, and Chuen-Tsai Sun

Posters

Analysing Problem Sequencing Strategies Based on Revised Bloom's Taxonomy Using Deep Knowledge Tracing 407
Sweety Agrawal and Amar Lalwani

MetaMentor: A System Designed to Study, Teach, Train, and Foster Self-regulated Learning for Students and Experts Using Their Multimodal Data Visualizations. 411
Roger Azevedo, Nicholas V. Mudrick, Michelle Taub, James Lester, Robert Taylor, Robert Sawyer, Kirby Culbertson, and Candice Roberts

What Can Eye Movement Patterns Reveal About Learners' Performance? . . . 415
Asma Ben Khedher, Imène Jraidi, and Claude Frasson

"Hypocrates": Virtual Reality and Emotions Analysis Towards a Personalized Learning 418
Marwa Boukadida, Hamdi Ben Abdessalem, and Claude Frasson

Embedding Speech Technology into Intelligent Tutoring Systems Using the CloudCAST Speech Technology Platform 421
André Coy, Phil Green, Stuart Cunningham, Heidi Christensen, José Joaquín Atria, Frank Rudzicz, Massimiliano Malavasi, and Lorenzo Desideri

Using E-learning System to Influence on User's Behavior Toward the Cybersecurity Strategy 425
Hasna Elkhannoubi and Mustapha Belaissaoui

Evolution of Methods of Evaluation in AI in Education. 428
Reva Freedman

Typing-Differences in Simultaneous Typed Chat 431
Michael Glass, Yesukhei Jagvaral, Chinedu Emeka, and Jung Hee Kim

Examining How Students' Typical Studying Emotions Relate to Those Experienced While Studying with an ITS 434
Jason M. Harley, François Bouchet, and Roger Azevedo

Examining How Typical Gaming Behavior Influences Emotions and Achievement During Gameplay 438
Jason M. Harley, Mohamed S. Benlamine, Maher Chaouachi, Claude Frasson, Yang Liu, and Aude Dufresne

The Design of a Learning Analytics Pedagogical Dashboard to Enhance Instructors' Facilitation in an Online Asynchronous Problem-Based Learning Environment 442
Lingyun Huang, Stephen Bodnar, Juan Zheng, Maedeh Assadat Kazemitabar, Yuxin Chen, Gurpreet Birk, Cindy E. Hmelo-Silver, and Susanne P. Lajoie

A Framework to Recommend Appropriate Learning Materials from Stack Overflow Discussions 446
Ashesh Iqbal, Mohammad Shamsul Arefin, and Mohammad Ali Akber Dewan

What Does It Mean to Provide the Right Level of Support During Tutorial Dialogue? 450
Sandra Katz, Patricia Albacete, Irene-Angelica Chounta, Pamela Jordan, Dennis Lusetich, and Bruce M. McLaren

Building Student Models in a Non-scaffolded Testing Environment. 454
H. Nguyen and C. W. Liew

Applying Human-Agent Team Concepts to the Design of Intelligent Team Tutoring Systems 457
Kaitlyn Ouverson, Mariangely Iglesias Pena, Jamiahus Walton, Stephen Gilbert, and Michael Dorneich

Concept-Based Learning in Blended Environments Using Intelligent Tutoring Systems 460
Ines Šarić, Ani Grubišić, Slavomir Stankov, and Timothy J. Robinson

Diagnosing Reading Deficiencies of Adults with Low Literacy Skills in an Intelligent Tutoring System 463
Genghu Shi, Andrew J. Hampton, Su Chen, Ying Fang, and Arthur C. Graesser

Everycoding: Combination of ITS and (M)OOC for Programming Education 466
Dongeun Sun and Hyeoncheol Kim

Adaptive Virtual Tutor Based on the Inference of the Student's Memory Content 470
Joanna Taoum, Elisabetta Bevacqua, and Ronan Querrec

Preliminary Evaluation of a Serious Game for Socio-Moral Reasoning. 473
Ange Tato, Aude Dufresne, Roger Nkambou, Frédérick Morasse, and Miriam H. Beauchamp

iMoodle: An Intelligent Moodle Based on Learning Analytics 476
Ahmed Tlili, Fathi Essalmi, Mohamed Jemni, Maiga Chang, and Kinshuk

Doctoral Consortium

Analysis and Optimization of Brain Behavior in a Virtual Reality Environment 483
Hamdi Ben Abdessalem

Enhancing EFL Students' Collaboration in a Blended Learning Environment: A Design-Based Research 486
Zexuan Chen and Jianli Jiao

Leveraging Mutual Theory of Mind for More Human-Like Intelligent Tutoring Systems 490
Bobbie Lynn Eicher, David Joyner, and Ashok Goel

Effect of Learning Support System for Feature Words on Group Learning . . . 493
Shun Okuhara and Takayuki Ito

A New Approach to Testing Children with Autism Spectrum Disorder Using Affect. 496
Veronica Rivera

Online Course Design to Support Learning Analytics in Facilitating Personalized Academic Help 499
Hongxin Yan and Kinshuk

Workshops

C&C@ITS2018: International Workshop on Context and Culture in Intelligent Tutoring Systems 505
Valéry Psyché, Isabelle Savard, Riichiro Mizoguchi, and Jacqueline Bourdeau

Learning Analytics: Building Bridges Between the Education and the Computing Communities 507
Sébastien Béland, Michel Desmarais, and Nathalie Loye

Exploring Opportunities for Caring Assessments 508
Diego Zapata-Rivera and Julita Vassileva

Making Sense Out of Synchronous and Asynchronous Discourse in Education (SADES) 510
Nia Dowell, Andrew Hampton, Xiangen Hu, and Christopher Brooks

Optimizing Human Learning 511
Fabrice Popineau, Michal Valko, and Jill-Jênn Vie

Industrial Tracks

ITS Adaptive Instruction Systems (AIS) Standards Workshop 515

ITS Applications Workshop 516

Tutorials

Automating Educational Research Through Learning Analytics: Data Balancing and Matching Techniques 519
David Boulanger, Vivekanandan Kumar, and Shawn Fraser

Authoring, Deploying and Data Analysis of Conversational Intelligent Tutoring Systems 520
Xiangen Hu, Zhiqiang Cai, Keith Shubeck, Kai-Chih Pai, Arthur Graesser, Bor-Chen Kuo, and Chen-Huei Liao

Author Index 521

Full Papers

Programming Intelligent Embodied Pedagogical Agents to Teach the Beginnings of Industrial Revolution

Ivan Luis Feix Baierle and João Carlos Gluz(✉)

Post-Graduation Program in Applied Computing (PPGCA),
UNISINOS, São Leopoldo, Brazil
ibaierle@hotmail.com, jcgluz@unisinos.br

Abstract. Combination of Virtual Reality and Artificial Intelligence technologies offer very interesting possibilities for educational purposes, allowing to design creative, intelligent and dynamic 3D virtual learning environments. However, nowadays there are few programming environments and tools that support Artificial Intelligence and agent programming techniques to control virtual 3D avatars. Aiming to help in this question, this work introduces a logical programming environment, which extends Prolog with BDI and multi-agent programming concepts and is fully integrated with Virtual Reality technology. The paper shows how this programming environment was used to create an interactive, animated and intelligent virtual world, focused on teaching the beginnings of Industrial Evolution. This educational virtual world was positively evaluated through experiments carried out with simulated classes of History.

1 Introduction

Virtual worlds (or virtual environments) are systems capable of realistically simulating three-dimensional (3D) space, allowing people to perceive a visual space close to reality (a Virtual Reality or VR) [11]. Besides the simulation of physical objects, a virtual world contains physical representations of external users (the *avatars*), which can be controlled by these users. Avatars can also be controlled by artificial agents, but in this case they are commonly referred to as NPCs (Non Player Characters).

Although various definitions have arisen for artificial agents, this paper assumes that an artificial agent is a computer system located in a given environment, which is able to perform actions autonomously in order to meet their goals [19]. Individual agents can be designed and developed in several ways following distinct architectures, however, this work focuses only on agents built with a *BDI* architecture, which is based on a rational actor model with mental attitudes of *Belief*, *Desire* and *Intention* [1].

Currently there are several programming environments to design and program BDI agents, but none of them can be easily integrated in a virtual 3D

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 3–12, 2018.
https://doi.org/10.1007/978-3-319-91464-0_1

environment to support real-time control of NPCs. From the point of view of Artificial Intelligence (AI) these kind of virtual worlds offer an exciting and complex environment to study and develop full incorporated (or *embodied*) agents in realistic physical and social environments, but, which are more controllable and do not suffer from real world physical problems that affect robots programming, such as mechanical problems with actuators or noise/interference in sensors. Think about the interesting possibilities of a Turing test conducted in a virtual world, where an agent (artificial or not) controlling an avatar must convince a human judge (simply another avatar) that he/she (or it) is not artificial.

This is the main technological motivation behind the present work, which introduces a logical programming environment, based on BDI model, which allows to design and program real-time animated agents that operate as NPCs in 3D virtual worlds compatible with the OpenSimulator (http://opensimulator.org). This environment, called *VirtuaLog*, extends the Prolog language, including logical abstractions to represent the concepts of the BDI model, multi-agent communication mechanisms and perception/action in three-dimensional virtual worlds.

The usefulness of this programming environment, is exemplified by its application to create a Social Sciences educational application focused on teaching the beginnings of Industrial Evolution. History is an important and particularly difficult area of application for intelligent educational systems, due to the complexity of natural language interactions that artificial agents need to master to become good pedagogical agents in this area. VR technology also has the potential to introduce the student into a fictional (but realistic) reality, very appropriate for the teaching of History.

The Watt virtual world can be considered as an alternative to the traditional way of teaching History, helping in teaching about the social, economic, scientific and technological processes that occurred during the initial phase of Industrial Revolution. The technology of intelligent pedagogical agents is used to control NPCs representing James Watt, as well as other characters characteristic of the mines and factories of the time, capable of interacting with students and teachers in a language close to natural, in order to solve doubts, propose missions and explain content related to the scenario. Basic gamification techniques are used, making the students undertake missions of recognition and collection of information in the scenario, for later analysis and reflection. The Watt world is open source and can be downloaded in http://obaa.unisinos.br/virtualog/.

2 Related Work

There are few programming environments and tools that support artificial intelligence and agent programming techniques to create and control virtual 3D avatars. The most effective systems to date combine web languages like WebGL [12], X3D [2] and AIML [17] to program virtual 3D chat-bots based on web formats and protocols. However, this kind of solution does not support full 3D worlds running on OpenSimulator or other kind of 3D engine, like Unity. In general, most of virtual reality programming today for these simulators and 3D engines is being made in traditional languages like Java, C++ or C# or script languages like LSL or Python [7,13]. There are, of course, several agent programming environments like JASON [1] and JACK [18] that support the BDI model, but they are not compatible with programming frameworks that implement access to 3D virtual worlds simulators or engines.

More recently, platforms and systems to design human-like animation characters as artificial agents have been proposed. This is the case of ADAPT platform [15] and animation system proposed in [14]. But, although these works provide good support to low-level animation and allow the procedural modeling of agents behavior, they do not support high-level cognitive BDI modeling of agents behavior, neither do they offer an ontological abstract view of 3D environment, like the virtual agents programming introduced in present work. Indeed, this programming environment can be seen as an effective high-level implementation of the embodied autonomous agent model proposed in [4], combining full BDI support for agent reasoning with a high-level environmental ontology, which defines all perceptions and actions possible for agents in the 3D environment. Basic animation tasks like locomotion, touching, grabbing, facial animations and body gestures are left in charge of OpenSimulator and viewer software.

Although nowadays there is a reasonable quantity of VR material to teach historical facts [16], these are essentially VR content to be navigated and observed. There is no interaction with intelligent pedagogical NPCs in natural language inside this environments. The use of intelligent NPCs for educational purposes in virtual worlds it is a new technological challenge. There are a relatively few examples of this kind of work: the [5] work shows how to implement "virtual humans" to serve as guides to explain the history of buildings in a virtual world and the [6] presents a 19th-century Singapore world, where synthetic characters provide informational and conceptual scaffolding to students. Although these works have similar goals to the present work, they differ in important aspects. Agent cognition abstraction layer is cited but not implemented or modeled in [5] work, only the immediately inferior behavior layer is implemented as hierarchical finite state machines. The approach proposed in the present work, based on a logical BDI model offers a more advanced model for agent cognition. The work [6] uses a goal-oriented scheduler in Java to design its synthetic characters, but has no correlated logical or ontological view about virtual world entities. None of these papers comments on how natural language support will be done or if it will be offered.

3 Programming NPCs in Prolog

Figure 1 presents the layered architecture of VirtuaLog programming environment used to build the educational virtual world presented in this work. The Virtual Agents Layer implements high-level agent programming abstractions represented by Prolog operators and predicates. The main component of this layer is the Virtual Agents Management system, which manages the connection of a Prolog agent with the corresponding NPC in a virtual world and is integrated with three subsystems: *SymPAL* (Symbolic Perception-Action Logic) subsystem that provides the logical interface of actions and perceptions of the NPC in the environment; *AgILog* (Agent Illocutionary Programming Logic) subsystem, which extends Prolog with BDI abstractions like beliefs and goals, supports high-level inter-agent communication and allows goal parallelism through threads, transforming Prolog in an agent programming similar to AgentSpeak (L) [1]; *DiaLogic* (Dialog Interaction Logic) subsystem, which provides a logical programming interface to create chat systems similar to AIML in Prolog. All of these systems and subsystems are open-source software developed by our research group and can be downloaded at http://obaa.unisinos.br/virtualog/.

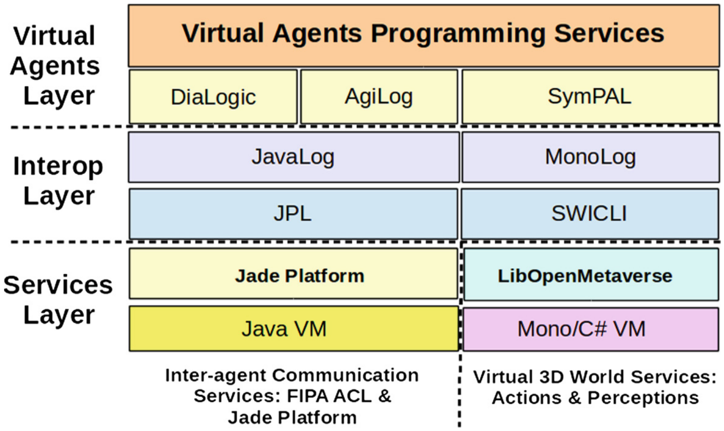

Virtual Agents Layer
Virtual Agents Programming Services
DiaLogic
AgiLog
SymPAL
Interop Layer
JavaLog
MonoLog
JPL
SWICLI
Services Layer
Jade Platform
LibOpenMetaverse
Java VM
Mono/C# VM
Inter-agent Communication Services: FIPA ACL & Jade Platform
Virtual 3D World Services: Actions & Perceptions

Fig. 1. Virtual agents programming environment architecture

An agent is created by $start\_agent(AgId, MainGoal)$ predicate where *AgId* is an atom that identifies the agent and *MainGoal* is a Prolog query that defines the main goal to be achieved by the agent. After this call, a new agent is created with *MainGoal* as the predicate called in the main thread of this agent. The plans to achieve these goals are *logical plans* composed of Prolog rules: it is the Prolog inference engine that transforms the logical plan in an operational plan. Besides plans, agents can have beliefs, which are Prolog facts stored in a base of facts specifically associated with the agent. The '++', '−' and '−+' operators allow, respectively, the addition, deletion and modification of beliefs, which are literal terms of Prolog.

An agent could have other goals linked to events that may occur in its belief base or be perceived in the environment. Such event-related goals are specified through the $handle\_event(EvPatt, EvHandler)$ predicate, which associates the event pattern *EvPatt* to the *EvHandler* goal. It is possible to wait for an event for a period of time using $wait\_event(EvPatt, Timeout)$ predicate. Events can be generated by the programmer through $signal\_event(Event)$ predicate or they can be generated by some subsystem. The AgiLog subsystem generates events related to the modification of belief base and reception of messages from other agents. The SymPAL subsystem generates events related to virtual world, including reception of textual messages from avatars and collisions between NPC and virtual world objects.

The connection of a Prolog agent with its corresponding NPC is done through *connect_avatar*(*First*, *Last*, *Pass*, *URL*) predicate called from some goal of controlling agent. After establishing the connection, actions can be performed by the avatar through ':>' operator. These actions were classified in SymPAL ontology by following categories: (a) Motion: actions to move NPC position and modify its physical arrangements (sitting, driving it to some other avatar, etc.). (b) Observation: actions to obtain information about the NPC, and about other objects and avatars. (c) Modification: actions to create, move and rotate objects, in addition to changes in dimensions, colors and textures of objects. (d) Personal: actions to modify NPC aspects, as clothing and body characteristics. (e) Interaction: operations to communicate with other avatars. The result of any observation action is a set of perceptions automatically added to the agent's perceptions base, which is a part of agent's belief base. Queries to perceptions base are made with '<:' operator applied to a term that represents a perception.

Entities perceived in virtual world have Universally Unique IDentifiers (UUID) to distinguish them one from another. The SymPAL ontology defines a logical representation for each kind of entity that can be seen in an OpenSimulator virtual world (see Fig. 2).

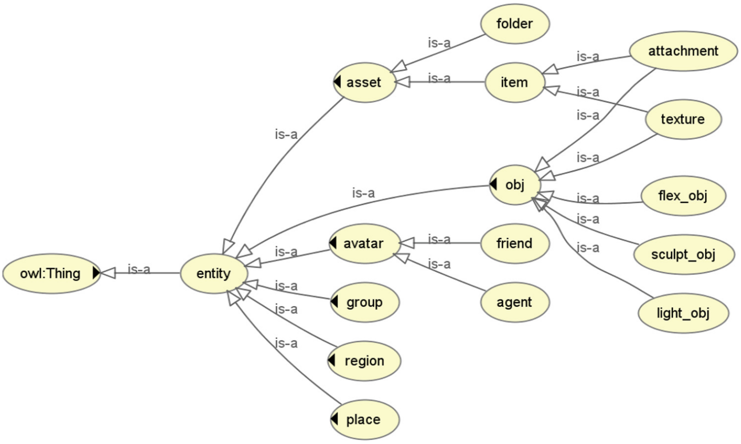
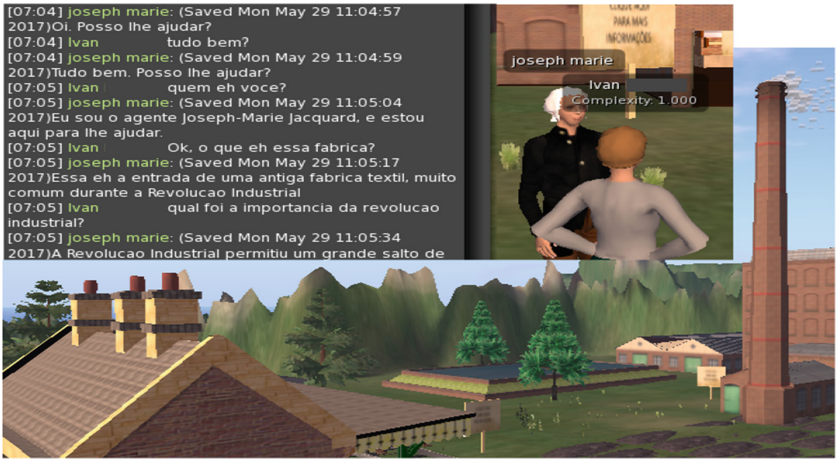


Fig. 2. Categories of perceptions in SymPAL ontology

4 Industrial Revolution Virtual World

The Industrial Revolution is considered one of the greatest technological leaps ever made in history. Its emergence occurred in England, and was known for the transition from the old manufacturing processes to the machining processes [3]. It was marked by the emergence of three technological innovations that shaped social, economic and cultural aspects of society [8]: (a) the adoption of machines in fabric manufacturing, (b) the generalization of the steam engine in practically all segments, and (c) large-scale production of iron using coal. The Watt world was designed as an interactive, animated and self-explanatory virtual scenario organized around these three major technologies. The scenario exposes technological artifacts (engines, looms and steam engines) and simulated spaces of factories and coal mines existing at the beginning of revolution. It is composed by scenes, each one integrated by a characteristic technological artifact. Scenes have animated objects responsible for providing information and interaction with students. In addition, some scenes feature a character of the era embodied as an NPC who is controlled by an intelligent pedagogical agent able to explain through natural language interactions, what the artifact is, what it can do and what it its importance for the historical period. Interactions occur through the instant messaging service. Figure 3 shows an example of interaction with Joseph-Marie Jaquard NPC in the scene four, which explores the emergence the mechanization of textile industry, revolutionizing the way spinning and weaving machines worked.



```
[07:04] joseph marie: (Saved Mon May 29 11:04:57
2017)Oi. Posso lhe ajudar?
[07:04] Ivan            tudo bem?
[07:04] joseph marie: (Saved Mon May 29 11:04:59
2017)Tudo bem. Posso lhe ajudar?
[07:05] Ivan            quem eh voce?
[07:05] joseph marie: (Saved Mon May 29 11:05:04
2017)Eu sou o agente Joseph-Marie Jacquard, e estou
aqui para lhe ajudar.
[07:05] Ivan            Ok, o que eh essa fabrica?
[07:05] joseph marie: (Saved Mon May 29 11:05:17
2017)Essa eh a entrada de uma antiga fabrica textil, muito
comum durante a Revolucao Industrial
[07:05] Ivan            qual foi a importancia da revolucao
industrial?
[07:05] joseph marie: (Saved Mon May 29 11:05:34
2017)A Revolucao Industrial permitiu um grande salto de
```

Fig. 3. Example of interaction with Joseph Marie Jaquard pedagogical NPC

Figure 4 shows the architecture and organization of the software implementing the Watt virtual world. OpenSim simulator is the basis of this world, being responsible for managing the visual data generated during the simulation. Control and management of educational NPCs running in this world is made by intelligent pedagogical agents developed in Prolog with the virtual agents programming environment introduced in Sect. 3. These agents communicate with

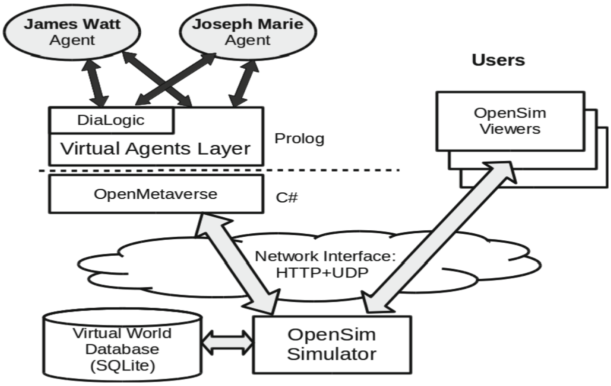
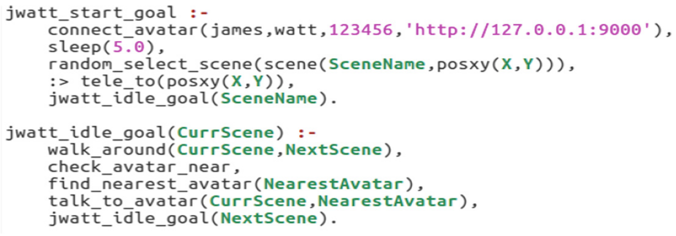


Fig. 4. Architecture and organization of the industrial revolution virtual world



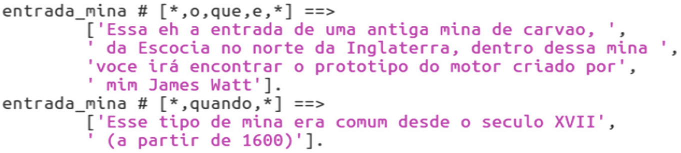
```
jwatt_start_goal :-
    connect_avatar(james,watt,123456,'http://127.0.0.1:9000'),
    sleep(5.0),
    random_select_scene(scene(SceneName,posxy(X,Y))),
    :> tele_to(posxy(X,Y)),
    jwatt_idle_goal(SceneName).

jwatt_idle_goal(CurrScene) :-
    walk_around(CurrScene,NextScene),
    check_avatar_near,
    find_nearest_avatar(NearestAvatar),
    talk_to_avatar(CurrScene,NearestAvatar),
    jwatt_idle_goal(NextScene).
```

Fig. 5. Main plan of James Watt agent

the OpenSim simulator using the same internet HTTP and UDP protocols that OpenSim-compatible viewers.

Figure 5 shows the main plan of James Watt agent, which begins with the connection to the NPC followed by a plan to move through the action points in the scene (*walk_around*() predicate), while it looks for avatars that are close and want to talk (*find_nearest_avatar*() and *talk_to_avatar*() predicates). The DiaLogic subsystem is responsible for interpreting dialog interaction rules, implementing in Prolog a dialog control mechanism similar to the AIML language [17]. Figure 6 shows a small sample of James Watt agent DiaLogic rules base.



```
entrada_mina # [*,o,que,e,*] ==>
        ['Essa eh a entrada de uma antiga mina de carvao, ',
        ' da Escocia no norte da Inglaterra, dentro dessa mina ',
        'voce irá encontrar o prototipo do motor criado por',
        ' mim James Watt'].
entrada_mina # [*,quando,*] ==>
        ['Esse tipo de mina era comum desde o seculo XVII',
        ' (a partir de 1600)'].
```

Fig. 6. Example of dialog rules of James Watt agent

5 Experiments and Results

The Watt virtual world was evaluated by laboratory and field experiments. Laboratory experiments verified functionality, stability, and performance of the prototype of this world. They also evaluated applicability of low-cost immersion technology to access virtual world. First laboratory experiment used a non-immersive desktop viewer to verify navigation of all scenes and check conversational functionality, stability and responsiveness of pedagogical agents. This experiment shown that performance of updating the 3D visualization, responsiveness of animations and textual interactions (by IM message) was appropriate. Second experiment used 3D immersion through low-cost (aka card-board) smartphone-based RV glasses with navigation by game controller. This experiment shown that low cost solution has sufficient quality for educational purposes for up to half an hour of use.

Main goal of field experiment was to make an evaluation of pedagogical impact of virtual's world use. This experiment also verify usability degree of the world. These experiments were carried out with a convenience sample formed by fourteen people, with five subjects having high school, four doing undergraduate and five with graduation. The average age of subjects were 26 years. Participants received an explanation about virtual world and afterwards had to carry out the virtual world exploration mission.

Pedagogical impact was evaluated by an experiment using a pre-test/post-test approach: all fourteen subjects answered questionnaires about the contents of History worked in the virtual world, before and after the experience of use. Each test was composed of ten randomly selected questions about social, economic and technological consequences of Industrial Revolution. Results of this experiment are presented in Fig. 7.

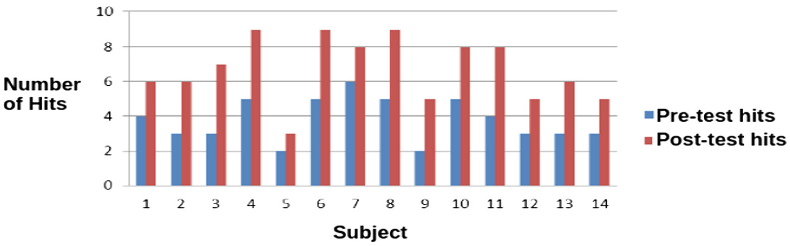


Fig. 7. Results of pedagogical experiment

Average pre-test score was 37.85% and average score of the post-test went up to 67.14%, providing good evidence that use of virtual world aided in learning about the Industrial Revolution. A Wilcoxon paired non-parametric test was used to assess whether difference between pre and post-test is significant, because this is the recommended test when samples did not necessarily have a normal distribution [9]. The Wilcoxon test applied to data from educational experiment resulted in a p-value of 0.0004886. This test indicates that the null hypothesis, which the use of virtual world did not increase the mean, can be rejected at a level of significance lower than 5%, that is, when p-value <0.05. Thus the alternative hypothesis that use of this world contributes to the increase of percentage of correct answers can be accepted.

The usability experiment followed TAM2 methodology [10]. After post-test, subjects of pedagogical evaluation experiment answered an usability questionnaire about Watt world. Results of this experiment are presented in Fig. 8, which shows average Likert scale obtained in respect to assertions designed to evaluate Perceived Easiness of Use (items 1 to 4 in Fig. 8) and Perceive Usefulness (items 5 to 10 in Fig. 8) variables of TAM2. These results are good evidence that usability achieved a high index of satisfaction.

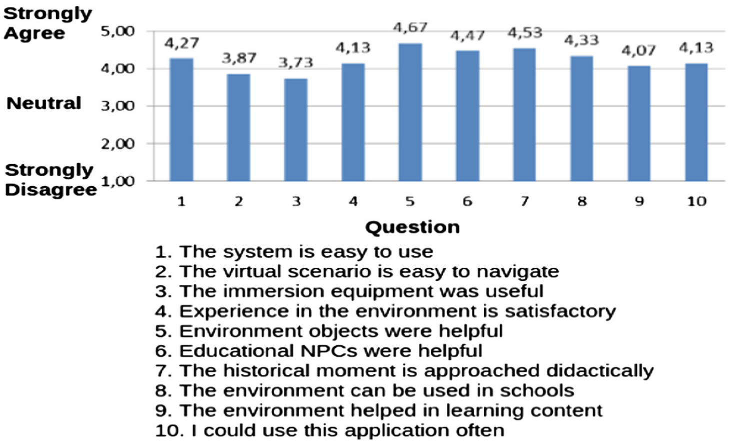


Fig. 8. Results of usability experiment

6 Conclusions

The VirtuaLog programming environment introduced in this paper allowed one programmer to design, develop and test the Watt virtual world in a period of 9 months. Note that this programmer have no previous experience with programming or designing 3D games or 3D applications, neither previous knowledge about logic programming. The previous experience of the programmer was only with Web programming languages and tools, such as Java, JavaScript and HTML5. Thus this feat can be considered as evidence of usefulness of VirtuaLog programming environment. Besides, initial experiments carried out with Watt world show evidences that its use can help to increase quality in the learning of History. The direction of the research now turns to advance the possibilities that natural language interface brings for educational purposes. The introduction of more gamification techniques is also an important objective, allowing more diversified and playful missions, containing different levels of challenges and awards.

References

1. Bordini, R., Hbner, J.F., Wooldridge, M.: Programming Multi-Agent Systems in AgentSpeak using Jason (Wiley Series in Agent Technology). Wiley, Hoboken (2007)
2. Brutzman, D., Daly, L.: X3D: Extensible 3D Graphics for Web Authors. Morgan Kaufmann Publishers Inc., San Francisco (2007)
3. Bynum, W.: A Little History of Science. Yale University Press, New Haven (2013)
4. Feng, A., Shapiro, A., Lhommet, M., Marsella, S.: Embodied autonomous agents. In: Handbook of Virtual Environments: Design, Implementation, and Applications, pp. 335–352 (2014)
5. Ieronutti, L., Chittaro, L.: Employing virtual humans for education and training in X3D/VRML worlds. Comput. Educ. **49**(1), 93–109 (2007). web3D Technologies in Learning, Education and Training
6. Jacobson, M.J., Miao, C., Kim, B., Shen, Z., Chavez, M.: Research into learning in an intelligent agent augmented multi-user virtual environment. In: Proceedings of IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, WI-IAT 2008, vol. 3, pp. 348–351. IEEE, Washington, DC (2008)
7. Kessler, G.D.: Virtual environment models. In: Handbook of Virtual Environments - Design, Implementation, and Applications, pp. 259–283 (2014)
8. Landes, D.S.: The Unbound Prometheus: Technological Change and Industrial Development in Western Europe from 1750 to the Present. Cambridge University Press, Cambridge (1969)
9. Malhotra, R.: Empirical Research in Software Engineering: Concepts, Analysis and Applications. CRC Press, Boca Raton (2016)
10. Marangunić, N., Granić, A.: Technology acceptance model: a literature review from 1986 to 2013. Univ. Access Inf. Soc. **14**(1), 81–95 (2015)
11. Messinger, P.R., Stroulia, E., Lyons, K., Bone, M., Niu, R.H., et al.: Virtual worlds - past, present and future: new directions in social computing. Decis. Support Syst. **47**(3), 204–228 (2009)
12. Parisi, T.: Programming 3D Applications with HTML5 and WebGL: 3D Animation and Visualization for Web Pages, 1st edn. O'Reilly Media Inc, Sebastopol (2014)
13. Rodrigues, N., Magalhes, L., Moura, J.P., Chalmers, A., Santos, F., Morgado, L.: Procedural virtual worlds. In: Zagalo, N., Morgado, L., Boa-Ventura, A. (eds.) Virtual Worlds and Metaverse Platforms: New Communication and Identity Paradigms, pp. 16–32. IGI Global (2012)
14. Shapiro, A.: Building a character animation system. In: Allbeck, J.M., Faloutsos, P. (eds.) MIG 2011. LNCS, vol. 7060, pp. 98–109. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-25090-3_9
15. Shoulson, A., Marshak, N., Kapadia, M., Badler, N.I.: Adapt: the agent development and prototyping testbed. In: Proceedings of the Symposium on Interactive 3D Graphics and Games (I3D), Orlando, FL. ACM (2013)
16. Stone, R.J., Hannigan, F.P.: Applications of virtual environments: an overview. In: Handbook of Virtual Environments, 2nd edn. (2014)
17. Wallace, R.: The Elements of AIML Style. Alice AI Foundation (2003)
18. Winikoff, M.: Jack™ intelligent agents: an industrial strength platform. In: Bordini, R.H., Dastani, M., Dix, J., Seghrouchn, A. (eds.) Multi-Agent Programming: Languages, Platforms and Applications, pp. 175–193. Springer, Boston (2005). https://doi.org/10.1007/0-387-26350-0_7
19. Wooldridge, M.: An Introduction to MultiAgent Systems. Wiley, Hoboken (2009)

Adaptive Clustering of Codes for Assessment in Introductory Programming Courses

Alexandre de A. Barbosa[1,3](✉), Evandro de B. Costa[2,3], and Patrick H. Brito[1,2]

[1] Federal University of Alagoas, Arapiraca Campus, Arapiraca, AL, Brazil
alexandre.barbosa@arapiraca.ufal.br

[2] Computer Science Institute, Federal University of Alagoas, Maceio, AL, Brazil
ebc.academico@gmail.com, patrick@ic.ufal.br

[3] Federal University of Campina Grande, Campina Grande, PB, Brazil

Abstract. Despite the importance of introductory programming disciplines, it is quite common to find problems related to academic students performance. In such environments, we easily find unmotivated students with some doubts and that do not understand basic programming concepts. Monitoring each of the students is not trivial because the number of students is high and, to do so, it would be necessary to observe many characteristics of each code submitted for practical activities. The teacher, even when helped by TAs (Teacher Assistants), is not able to perform the reviews quickly, for this activity requires a huge amount of time. Fast feedback is extremely important to enable the learning of any concept. In this research, we investigate an adaptive approach to cluster codes in order to minimize the effort of evaluation. The results vary from reasonable to perfect concordances, considering the semiautomatic evaluations obtained with the clustering and the expert evaluations.

1 Introduction

Programming is one of the basic competences in computer science, it is the basis for the development of several other competences required for professionals in the area. The initial periods of Computer Science (CS) courses in general encompass different disciplines that focus on the study of algorithms and the implementation of programs. In such environments, we easily find unmotivated students with some doubts and that do not understand basic programming concepts. In addition, many of the approved students do not have the necessary competencies for the course and professional life [1].

Practical coding activities are typically adopted in programming courses. Assessment of the proposed solutions is quite difficult. A large number of parameters can be used. To evaluate a code, in addition to identifying whether the correct outputs are generated for an input set, an evaluator can use efficiency, manutenability, documentation and other aspects. Given that the evaluation is time-consuming, it is subject to the bias and errors of each evaluator.

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 13–22, 2018.
https://doi.org/10.1007/978-3-319-91464-0_2

The student, without fast and adequate feedback, is unaware if their understanding is correct or not. In this way, possible doubts are not presented and new contents are given by the teacher without the previous ones had been fully understood. The teacher, even when helped by the TAs, is not able to perform the reviews quickly. When trying to provide a rapid assessment, it will probably lack quality, otherwise, when trying to provide high-quality feedback, there will be a work overhead and a delay in response.

In this scenario, it is not viable to have individualized help for each student. However, fast feedback is of extreme importance to enable the learning of any concept [2]. Thus, some researches have been developed with the aim to propose methods and tools to facilitate the monitoring of the activities of students in programming courses. Some of these researches, such as [3–5], suggests the use of peer assessment as a means of providing fast and effective feedback. This solution is broadly used in Massive Open Online Courses (MOOCs), as described in [6,7], where the courses are applied to hundreds or thousands of people enrolled in them, and just as occurs in the context of programming, it is impossible for the teacher to evaluate each solution.

During the evaluation of codes proposed as solution to a problem, the teacher observes similar solutions. However, it is unproductive to find for these similar solutions manually. In this research, to tackle part of the problem, one of its facets was investigated, concerning the possibility of clustering codes using features extracted only from the source codes. Besides that, knowing that different evaluators can adopt distinct assessment criterias. The clustering approach is adaptable to each evaluator. As main contribution of this research, it is expected that the clustering of codes can be used to allow the teacher to make fast feedback during the evaluation process and provide more detailed information to the students.

Two objectives were adopted in the research, which are: (1) identify if the semiautomatic evaluation obtained with the use of code clustering is similar to the evaluation of a specialist; (2) verify the concordance of the evaluations obtained with clustering in comparison with two human specialists.

This paper is organized as follows, in the next section some considerations about the assessment of codes in an introductory programing course are presented; In Sect. 3 some related works are shown; The proposed method of clustering codes is described in Sect. 4; Finally the last section presents the conclusions and further work related to the research.

2 Assessment in Introductory Programing Courses

Typically in programming courses practical coding activities are used as part of the learning process. In these activities the teacher generally creates a set of problems, and the students have to code a solution to each one of those problems. The teacher must select a set of exercises that makes possible to perform an appropriate assessment in a short time, because, as mentioned before, fast feedback is an important factor for learning.

However, the evaluation of these activities is time-consuming. One of the most common criterias adopted in evaluation of codes is the identification of whether the correct outputs are generated for an input set. The teacher can execute the code and check the outputs manually or this can be done by a tool using a previously defined test set. In addition, other code quality characteristics can be used in an evaluation, these may vary from evaluator to evaluator. Given the difficulties related to the process, it is subject to the bias and errors of each evaluator.

A strategy that is used to make fast assessment is the division of work. Typically the teacher and a set of teacher assistants (TAs) take a subset of the solutions and evaluate them. This way, it is necessary that the evaluators specify a default manner to evaluate.

The assessment of codes can create a classification and a list of observations for each code. Typically a class (e.g. A, B, C, D) or a number (e.g. integers values from 0 to 10 or 0 to 100) is used to classify the solutions. A list of observations may comprehend the indication of where there were errors or some tips to produce better code. In this research only the classification aspect of the feedback was investigated. The classifications were given as an integer number from 0 to 10.

3 Related Work

Work related to the context of automatic generation of evaluations, automatic grading and automatic feedback, are not recent. The work of Hext and Winings [8], besides the work of Forsythe and Wirth [9], for example, date from the sixties. Many of the recent surveys continue to explore similar ideas to their works.

The vast majority of automatic code evaluators, among which we cite [10], use online judges, that is, the assessment is based on acceptance tests. Some of these proposals complement the indication of success, or failure, provided by the tests with tips that help the student to identify the errors. Such tips are associated with a test case based on the tester's experience in determining the likely cause of the failure.

Several researches, as an example we cite [11–13], use an analysis of similarities with other purposes that are not a detection of plagiarism. The works of Li et al. [13] and Biggers and Kraft [12] explore the code similarities from a Software Engineering perspective. The aim is, for example, to identify the adherence of code to functional requirements. In [11], it is investigated whether automatic evaluators (similarity algorithms) can compare student codes as well as specialists.

Other researches, among which we quote [14–16], have as objective to cluster or classify code solutions. A different set of techniques is used in each of these researches. Choudhury e Yin [15,16] uses the ABC metric [17], the *OPTICS* clustering algorithm and distance tree similarity measure, in order to aid, through tips, students enrolled in Software Engineering courses to produce better code. In Srikant's work, [14], machine learning techniques (linear regression, random forests and Support Vector Machines) are used to learn which properties are taken into account in the evaluation of a programming exercise.

The main contribution in the research described in this paper is to minimize the evaluation effort, considering the different criterias of each evaluator. Other similar works use fixed criterias, that is, it is assume that all evaluators use the same evaluation parameters.

4 Adaptive Clustering of Codes

4.1 Clustering Method

Clustering algorithms cluster the elements taking into account their similarity. In this way, it is correct to affirm that the elements of the same group are more similar to each other than to the elements of other groups. A software metric represents a way of verifying that a software has a given property. A software metric can be used with distinct objectives.

The clustering algorithm Kmeans [7] was used to cluster the sets of codes. Kmeans used a set of software metrics [18] as a way of comparison between codes. The following metrics were adopted: degree of adherence to a test set [18], ciclomatic complexity [19], a subset of Halstead metrics [20] (distinct operands and distinct operators), Jaccard similarity [11], text edit distance [11] and tree edit distance [11].

Kmeans is an unsupervised learning algorithm. There are ‘K’ centroids that are used to define clusters. An element is considered to be in a cluster based on the distance to each centroid. Kmeans computes the centroid using the set of data from all elements there are present on the cluster. At each iteration of the algorithm the centroids are recalculated and a new association of the elements is performed on the clusters. Iterations occur until the centroids are not altered. In Fig. 1, adapted from [21], the running process of Kmeans is shown using elements as points and centroids as crosses.

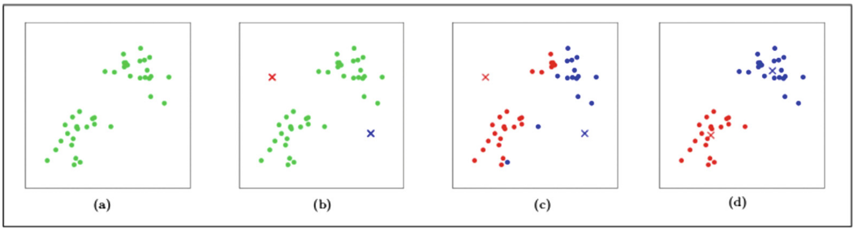


Fig. 1. K-means algorithm. Training examples are shown as dots, and cluster centroids are shown as crosses. (a) Original dataset. (b) Random initial cluster centroids. (c–d) Illustration of running two iterations of k-means [21].

4.2 Comparison Measures

To evaluate the proposed approach, two comparison measures were used, Cohen's Kappa [22] and euclidean distance. Cohen's Kappa coefficient is a statistical measure that can be used to compute the intensity of agreement between two lists of classifications. Euclidean distance can compute the distance of two points in an euclidean n-dimensional space.

Given two lists of classifications provided by two experts $E1$ and $E2$, we want to know how much $E1$ and $E2$ agree with each other. Cohen's Kappa coefficient computes the degree of agreement beyond what would be expected at random. Kappa (K) values can vary from -1 to 1. Negative values means that there are no concordance, with no interpretation for each value. Zero value means that there are concordance but it's exactally the degree of agreement there were expected at random. Positive values indicate concordance, with a degree of agreement, greatest the value is, greatest the concordance is. Kappa interpretation levels can be seen on Table 1. The null hipotesis that can be tested with Cohen's Kappa is that there is no concordance in two lists of classifications.

Table 1. Cohen's Kappa coefficient interpretation

Interval	$(0; 0.2)$	$[0.2; 0.4)$	$[0.4; 0.6)$	$[0.6; 0.8)$	$[0.8; 1.0)$	1
Agreement interpretation	Slight agreement	Fair agreement	Moderate agreement	Strong agreement	Almost perfect agreement	Perfect agreement

Euclidean distance represents the distance between two points in a n - dimensional space. This way, we can use represent a point in an n - dimensional using the list of grades. Then, the lower the distance value is, the smaller the total difference in the list of grades. When euclidean distance is 0 (zero) it can be interpreted as a perfect match in the grades lists, or the specialists gave exactly the same grades for each code. Other values of euclidean distance can be only interpreted as a comparison measure between each different solution. Therefore, there is no objective interpretation of the distance if we don't consider the relation with other distance values. Euclidean distance was a way to compare the efficiency of the approach for the different exercises.

Using these measures two kinds of comparisons were done, a comparison between the list of grades of two specialists, and a comparison of a list of grades obtained with the clustering approach and the list of grades of a specialist. Cohen's Kappa measure indicated the degree of agreement between the grades. In this measure a concordance occurs only when exactly the same grade is provided in the two lists. This justifies the use of euclidean distance. Using this measure it is possible to observe how much the list of grades are distant.

4.3 Adaptive Clustering

The adaptive clustering codes approach proposed in this paper is accomplished in four steps: (1) code metrics extraction (2) identification of the criteria adopted by the specialist (3) Clustering generation (4) Evaluation. Code metrics extraction correspond to the computation of each software metric used as in the property vector given for the clustering algorithm. Similarity metrics were extract only for one reference solution. The reference solution may be given by the teacher or selected in the set of solutions of the students.

For the identification of the criteria adopted by the specialist a brute force method was used. A set of each possible combination of the software metrics were created. We can interpret this as an oracle that can identify the evaluator criteria. Different techniques can be investigated to implement this oracle.

The combination of all metrics aimed to capture, with some degree of approximation, the criteria adopted by the specialists. As an example, if a specialist observes only correct outputs for the inputs provided, his criterion would be captured by an element were only the metric that measures the degree of adherence to a test set is used. Another specialist can observe the correct outputs for the inputs in combination with another code characteristic, for this, a different element on the set of combination of metrics will be identified.

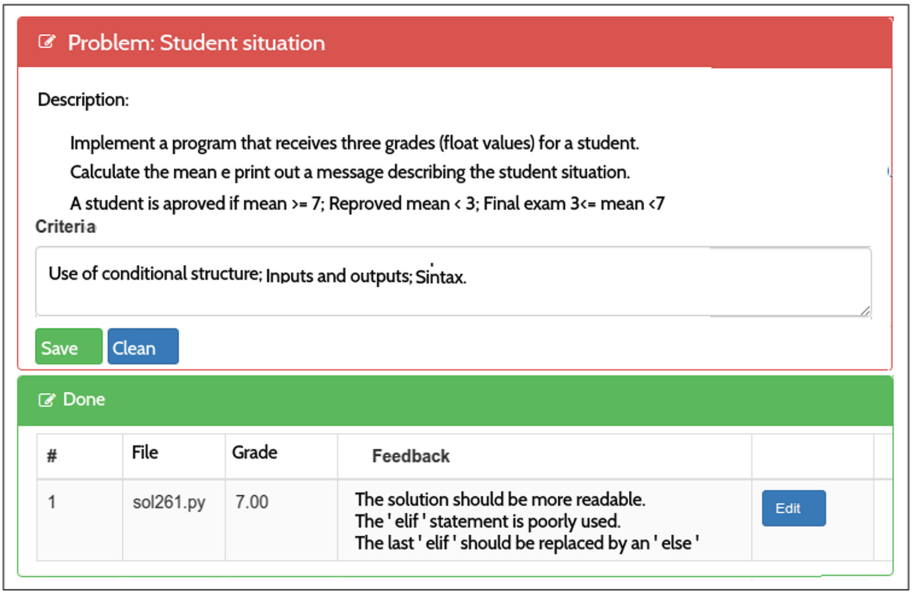
Clustering codes generation was done by using Kmeans, using the values $k = 5$ and $k = 10$. The choice of these values is justified because the evaluated code sets had a minimum of 23 submissions, and in the worst case, the evaluation effort would be reduced by approximately half. In all executions, default parameters for the Kmeans algorithm were used. For each possible property vector generated in the previous step, a cluster for each k value was created. From this set, the cluster with the highest degree of agreement in relation to the expert based on the evaluations provided will be selected.

The specification of evaluations occurred so that each specialist assigned a grade (classification) to a representative element of the group. This element is the one that has the smallest distance from the reference solution. This note is then generalized to all elements of the same group. Thus, when kmeans was executed by providing $k = 5$ only five expert evaluations were required. Similarly for $k = 10$ ten expert evaluations were required.

4.4 Methodology

The dataset used in this research consists of a set of programming problems (exercises), a set of codes submitted by students as solutions to the exercises, and a set of evaluations (grades varying from 0 to 10) provided by experts.

The set of exercises consists of six problems, these are: ‘salary bnus’ (32 submissions); ‘points distance’ (23 submissions); ‘student situation’ (43 submissions); ‘elections’ (40 submissions); ‘odd loop’ (41 submissions); and ‘divisible by 3’ (44 submissions). All submissions where from groups of students enrolled in the equivalent programing courses on different federal institutions of Northeast of Brazil. The set of codes submitted by the students consists of a total of 223 submissions of proposed solutions, or attempts to solutions. All codes were written in Python. Figure 2 shows the description screen there were presented in ‘student situation’ exercise.



Problem: Student situation

Description:

Implement a program that receives three grades (float values) for a student.
Calculate the mean e print out a message describing the student situation.
A student is aproved if mean >= 7; Reproved mean < 3; Final exam 3<= mean <7

Criteria

Use of conditional structure; Inputs and outputs; Sintax.

Done

#	File	Grade	Feedback	
1	sol261.py	7.00	The solution should be more readable. The ' elif ' statement is poorly used. The last ' elif ' should be replaced by an ' else '	Edit

Fig. 2. Description of a problem for a specialist and listing of performed evaluations.

The set of assessments was provided by 8 specialists, 2 of whom were teachers from a federal university, 6 were assistants in previous programing courses. Each specialist performed the evaluation procedures in the time and environment they judged most appropriate. When conducting an evaluation it was necessary to provide: (a) criteria description (as shown in Fig. 2); (b) a grade, from 0 to 10 (as shown in Fig. 2); (c) textual information for the students that submitted the code (as shown in Fig. 2); and (d) Any observations to the researchers, describing any problem that could occur.

Observing the descriptions of the adopted criterias it was possible to identify that different experts adopts different sets of criterias in the evaluation of the same problem. In the same way a specialist adopted different criteria in relation to the different problems evaluated.

4.5 Results and Discussion

The results were computed based on the maximum, minimum and average values for the comparative measures adopted considering each problem and specialist. The best results were obtained when Kmeans algorithm ran with ‘K = 10’. A general view of the results in shown in Table 2.

Thus, in order to identify if the semiautomatic evaluation obtained with the use of clustering codes approach is similar to the evaluation of a specialist, considering the average, a substantial agreement ($kappa$ = 0.76) was found. This result corroborates the values obtained with respect to the Euclidean distance.

Table 2. General results for clustering codes approach

Cohen's Kappa			Euclidian dist.		
Max.	Min.	Mean	Max.	Min.	Mean
1.00	0.39	0.76	19.57	0.00	5.95

An average value of 5.95, was found, which indicates a proximity between the evaluations. Thus, given the concordances, it is possible to suggest that the clustering approach provides as evaluation that is similar to the specialists' evaluation.

The graph shown in the Fig. 3 contains the boxplots related to the concordances obtained with the clustering approach when compared to experts and the general agreement for each expert with respect to their peers considering all the evaluations carried out.

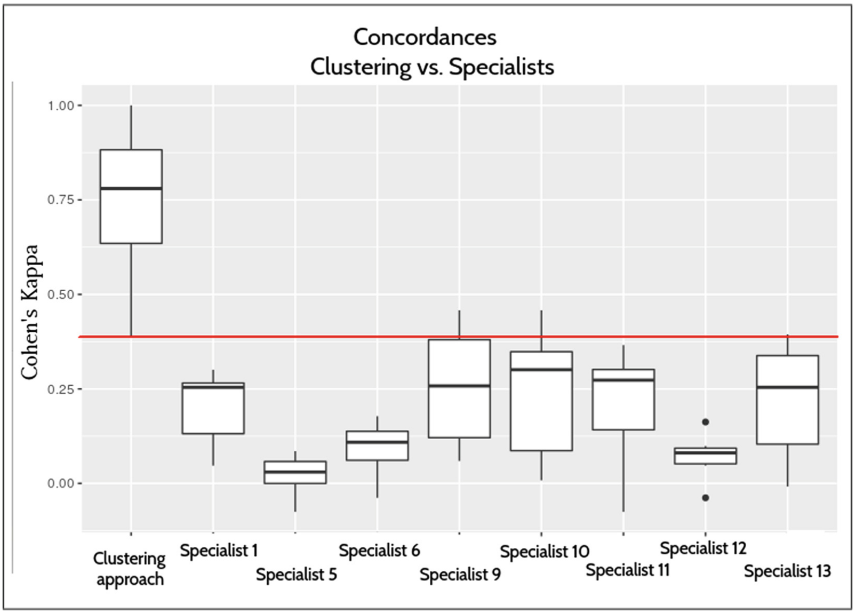


Fig. 3. Concordances using the approach and concordances between specialists.

Observing the concordance graph, it is possible to notice that the maximum value of agreement obtained between specialists (moderate concordance, $kappa = 0.45$) is very close to the minimum value obtained. The horizontal line that cuts the graph shows the relation between the minimum agreement of the cluster-based assessment (reasonable agreement, $kappa = 0.39$) in relation to the maximum agreement among specialists.

Considering the means, that is, how much a specialist agrees with his/her peers and how much the grouping approach generates evaluations with agreement with the expert, agreement between experts (reasonable agreement, $kappa = 0.2539$) was much lower than the agreement obtained with the clustering approach (substantial agreement, $kappa = 0.70$). Thus, it may be suggested that the cluster-based assessment provides a higher concordance than that obtained between two specialists.

5 Conclusions and Further Work

In this paper we have proposed the use of a clustering algorithm to minimize the effort expended in the evaluation of codes in introductory courses. The results suggest that it is possible to minimize the evaluation effort expended. Two observations support this assertion. An evaluator's criteria seems to be captured by software engineering metrics with a fair degree of accuracy. In addition, the evaluations of similar solutions were very close to the observed scenarios.

This research is an ongoing work, much investigation is still necessary. Taking as example, the comparison of different clustering techniques, the use of a larger set of metrics, the generation of textual feedback and other. Observing the success of other techniques in related researches, we expect the improvement of the proposed approach.

References

1. McCracken, M., Almstrum, V., Diaz, D., Guzdial, M., Hagan, D., Kolikant, Y.B.D., Laxer, C., Thomas, L., Utting, I., Wilusz, T.: A multi-national, multi-institutional study of assessment of programming skills of first-year CS students. In: Working Group Reports from ITiCSE on Innovation and Technology in Computer Science Education, ITiCSE-WGR 2001, pp. 125–180. ACM, New York (2001)
2. Stegeman, M., Barendsen, E., Smetsers, S.: Towards an empirically validated model for assessment of code quality. In: Proceedings of the 14th Koli Calling International Conference on Computing Education Research, Koli Calling 2014, pp. 99–108. ACM, New York (2014)
3. de Raadt, M., Toleman, M., Watson, R.: An evaluation of electronic individual peer assessment in an introductory programming course. In: Lister, R., Simon (eds.) Seventh Baltic Sea Conference on Computing Education Research (Koli Calling 2007), Koli National Park, Finland. CRPIT, vol. 88, pp. 53–64. ACS (2007)
4. Sitthiworachart, J., Joy, M.: Computer support of effective peer assessment in an undergraduate programming class. J. Comput. Assist. Learn. **24**(3), 217–231 (2008)
5. Warren, J., Rixner, S., Greiner, J., Wong, S.: Facilitating human interaction in an online programming course. In: Proceedings of the 45th ACM Technical Symposium on Computer Science Education, SIGCSE 2014, pp. 665–670. ACM, New York (2014)
6. Kulkarni, C., Wei, K.P., Le, H., Chia, D., Papadopoulos, K., Cheng, J., Koller, D., Klemmer, S.R.: Peer and self assessment in massive online classes. ACM Trans. Comput.-Hum. Interact. **20**(6), 33:1–33:31 (2013)

7. Piech, C., Huang, J., Chen, Z., Do, C.B., Ng, A.Y., Koller, D.: Tuned models of peer assessment in MOOCs. CoRR abs/1307.2579 (2013)
8. Hext, J.B., Winings, J.W.: An automatic grading scheme for simple programming exercises. Commun. ACM **12**(5), 272–275 (1969)
9. Forsythe, G.E., Wirth, N.: Automatic grading programs. Commun. ACM **8**(5), 275–278 (1965)
10. Yulianto, S.V., Liem, I.: Automatic grader for programming assignment using source code analyzer. In: 2014 International Conference on Data and Software Engineering (ICODSE), pp. 1–4. IEEE (2014)
11. Gaudencio, M., Dantas, A., Guerrero, D.D.: Can computers compare student code solutions as well as teachers? In: Proceedings of the 45th ACM Technical Symposium on Computer Science Education (2014)
12. Biggers, L.R., Kraft, N.A.: Quantifying the similiarities between source code lexicons. In: Proceedings of the 49th Annual Southeast Regional Conference, ACM-SE 2011, pp. 80–85. ACM, New York (2011)
13. Li, S., Xiao, X., Bassett, B., Xie, T., Tillmann, N.: Measuring code behavioral similarity for programming and software engineering education. In: Proceedings of the 38th International Conference on Software Engineering Companion, ICSE 2016, pp. 501–510. ACM, New York (2016)
14. Srikant, S., Aggarwal, V.: A system to grade computer programming skills using machine learning. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2014, pp. 1887–1896. ACM, New York (2014)
15. Choudhury, R.R., Yin, H., Moghadam, J., Chen, A., Fox, A.: Autostyle: scale-driven hint generation for coding style. In: Proceedings of the 13th International Conference on Intelligent Tutoring Systems, ITS, vol. 201, pp. 122–132 (2016)
16. Yin, H., Moghadam, J., Fox, A.: Clustering student programming assignments to multiply instructor leverage. In: Proceedings of the Second (2015) ACM Conference on Learning @ Scale, L@S 2015, pp. 367–372. ACM, New York (2015)
17. Fitzpatrick, J.: More C++ Gems, pp. 245–264. Cambridge University Press, New York (2000)
18. Sommerville, I.: Software Engineering, 9th edn. Addison-Wesley Publishing Company, Boston (2010)
19. McCabe, T.J.: Cyclomatic complexity and the year 2000. IEEE Softw. **13**(3), 115–117 (1996)
20. Halstead, M.H.: Elements of Software Science (Operating and Programming Systems Series). Elsevier Science Inc., New York (1977)
21. Piech, C.: K means. http://stanford.edu/cpiech/cs221/handouts/kmeans.html
22. Cohen, J.: Weighted kappa: nominal scale agreement with provision for scaled disagreement or partial credit. Psychol. Bull. **70**, 213–220 (1968)

Recommendation in Collaborative E-Learning by Using Linked Open Data and Ant Colony Optimization

Samia Beldjoudi[1,2](✉), Hassina Seridi[2], and Nour El Islem Karabadji[1,2]

[1] Superior School of Industrial Technologies, Annaba, Algeria
{s.beldjoudi,n.karabadji}@epst-annaba.dz
[2] Laboratory of Electronic Document Management LabGED, Badji Mokhtar University, Annaba, Algeria
seridi@labged.net

Abstract. Social tagging activities allow the wide set of web users, especially learners, to add free annotations on educational resources to express their interests and automatically generate folksonomies. Folksonomies have been involved in a lot of recommendations approaches. Recently, supported by semantic web technologies, the Linked Open Data (LOD) allow to set up links between entities in the web to join information in a single global data space. This paper demonstrates how structured content accessible via LOD can be leveraged to support educational resources recommender in folksonomies and overcome the limited capabilities to analyze resources information. Another limitation of resources recommendation is the content overspecialization conducting in the incapacity to recommend relevant resources diverse from the ones that learner previously knows. To address these issues, we proposed to take advantage of the richness of the open and linked data graph of DBpedia and Ant Colony Optimization (ACO) to learn users' behavior. The basic idea is to iteratively explore the RDF data graph to produce relevant and diverse recommendations as an alternative of going through the tedious phase of calculating similarity to attain the same goal. Using ant colony optimization, our system performs a search for the appropriate paths in the LOD graph and selects the best neighbors of an active learner to provide improved recommendations. In this paper, we show that ACO also in the problem of recommendation of novel diverse educational resources by exploring LOD is able to deliver good solutions.

Keywords: Folksonomies · E-learning · Recommendation · Linked Open Data Diversity · Ant Colony Optimization

1 Introduction

Social tagging systems have known a big success over the web in the last years, especially in recommendations approaches. This is due to the increasing number of users through those systems which caused the appearance of tag-based profiling approaches that support social recommendation.

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 23–32, 2018.
https://doi.org/10.1007/978-3-319-91464-0_3

A very precise recommender system could return a set of similar resources matching the user interests. The problem here is that the entire set of recommended resources may be obvious as one considers the case of a film recommendation algorithm that only returns films of the same actor. To overcome this problem, novelty and diversity should be also considered in the evaluation of a recommender system, as precision only offers an incomplete description of the system's effectiveness.

On the other hand, Linked Data designs data that following a model based on: using URIs to identify entities, using http-URIs so that users can look up them, giving valuable information at these URIs based on standard formats and connecting to other URIs so that users can discover more things (Berners-Lee 2007). For the data designed as Linked Open Data (LOD), it should as well be provided publicly, and be under an open license. LOD covers many fields like medicine, social networks, etc. Moreover, some data sources such as DBPedia give general, cross-domain information and so join data from very diverse fields.

Here we will present new approach to recommend educational resources applied in LOD by using Ant Colony Optimization (ACO). ACO is a computation algorithm from mimics the behaviors of ants' colony. Naturally, the behavior of one ant is simple contrary to the behavior of the whole ant colony which is very complicated; this is the case in our problem: the behavior of a single user is simple, but supervising all users and their tagged resources without calculating similarities is very complicated and therefore ACO is able to complete this difficult task. The key proposal of using ant colony metaphor is that ants are able to find their way from the nest to food and back for the reason that ant colony communicates via a chemical substance called pheromone. In our approach we have inspired from this idea to resolve our problem where we proposed to consider each user as an ant and try to him recommend novel diversified resources based on the LOD exploration and the feedback of the community. Knowing that, in the recommender system within LOD, the determination of the recommendation may be solved as a finding the best path in the oriented weighted graph.

The presented paper addresses different challenges in the social semantic web area. The main focus of our study is how to exploit the semantic aspect of LOD to enhance educational resource recommendation within social tagging activities. We propose a new method to analyzing learner profiles according to their tagging activity in order to improve resource recommendation. The effectiveness of recommendation depends on the resolution of social tagging drawbacks. In our recommender process, we demonstrate how we can assure diversity and novelty in recommendation by taking into account LOD exploration and ACO mimics in order to personalize recommendation. We used thus the force of linked open data to enhance educational resource recommendation in social tagging system by exploring the interlinked entities in LOD cloud. This paper is organized as follows: Sect. 2 is an overview of the main contributions related to our work. Section 3 is dedicated to the presentation of our approach. In Sect. 4 we present and discuss the results of some experiments we conducted to measure the performance of our approach. Conclusion and future works are described in Sect. 5.

2 Related Works

Social web based approaches, like folksonomies, have achieved a high level of improvement even in E-learning practice. In this section, an overview about the main contribution attached to this field is proposed.

In the paper of Torniai et al. (2008), the authors tried to propose a social semantic learning environment. The idea based on creating folksonomies from learners' tags and then helps to ontology maintenance. In Lau et al. (2015), the authors proposed to annotate learning resources with a lightweight annotation metadata schema complemented by a folksonomy-derived semantic model. According to the authors, the annotation metadata schema follows a novel strategy to associate numerical ratings to learners' subjective expression of opinions on learning resources. On the other hand, the semantic model serves to support a collaborative resource recommendation algorithm based on the k-nearest neighbor approach. Klašnja-Milićević et al. (2015) presented an overview of the most key requirements and challenges for applying a recommender system in e-learning environments. The authors presented the various limitations of the current approach of recommendation techniques and promising extensions with model for tag-based recommender systems, which can apply to e-learning environments in order to offer improved recommendations. In another contribution, (Kopeinik et al. 2017) investigated the application of two tag recommenders that are inspired by models of human memory. The authors find that displaying tags from other group members helps significantly in semantic stabilization in the group, as compared to a strategy where tags from the students' individual vocabularies are used. In Beldjoudi et al. (2016), the authors proposed a new approach for personalizing and improving resources retrieval in collaborative learning with tackling tags ambiguity and event detection impact on ranking retrieved resources. In another contribution (Beldjoudi et al. 2017) proposed a method to analyze user profiles according to their tags in order to predict interesting personalized resources and recommend them. The authors proposed a new approach to reduce tag ambiguity and spelling variations in the recommendation process by increasing the weights associated to web resources according to social similarities. They base upon association rules for discovering interesting relationships among a large dataset on the web. Karabadji et al. (2018) proposed to focus mainly on the growing of the large search space of users' profiles and to use an evolutionary multi-objective optimization-based recommendation system to pull up a group of profiles that maximizes both similarity with the active user and diversity between its members. According to the authors, in such manner, the recommendation system will provide high performances in terms of both accuracy and diversity. In our work we want to leverage the social and semantic web in order to enhance educational resources recommendation in collaborative e-learning.

3 Approach Description

In e-learning domain, we define folksonomy by a tripartite model where web resources are associated with a learner to a list of tags. Formally a folksonomy is a tuple F = <L, T, R, A> where L, T and R represent respectively a set of learners, a set of tags and a set of educational resources, and A represents the relationships between the three preceding elements, i.e. A ⊆ L × T × R.

In this paper, we propose a method to analyze learners' profiles according to their tags in order to predict interesting personalized resources and recommend them. The objective was to enrich the profiles of folksonomy learners with pertinent educational resources.

The effectiveness of the recommendation depends on the resolution of diversity and novelty problems. In our approach we tackle these drawbacks to enhance our recommender system. The detail will be described below.

3.1 LOD Exploration to Ensure Diversity and Novelty in Recommendation

When using a recommender system such as those of online stores, the results are mainly obvious and expected by the users. In this case, it is clear that the recommendation is not very helpful in the sense of the lack of diversity and novelty.

To solve this dilemma in folksonomies-based collaborative learning, we propose extracting the most popular features found in the resources-based learner profile (i.e. the characteristics that interest the learner when he tag his resources) and then explore the LOD to extract resource linked with these features. For example, let us consider the case presented in Fig. 1.

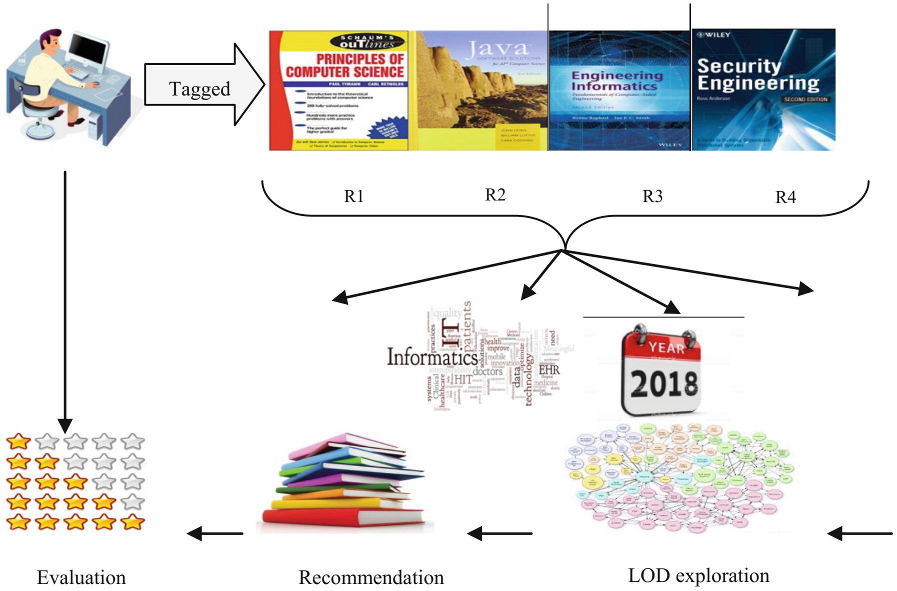


Fig. 1. The recommender system process

In this example, the profile of the learner is composed from the resources (R1, R2, R3 and R4), Thus the intersection between the resources' features must be calculated (R1 ∩ R2 ∩ R3 ∩ R4), this is done because we want to extract the most popular characteristics that interest the learner when he choose tagging his resources. Then for each feature (Pi) in the result of intersection we will explore the LOD graph in the first level to extract other resources (R5) having these features or having a direct/ indirect link with these later (R6, R7 resp).

We have in the above example (R1 ∩ R2 ∩ R3 ∩ R4) = {[domain: informatics]; [author: …]; [year: …]; [edition: …]…}. By exploring the LOD graph we find that the resource "informatics" is linked with other resources (for example: "University, Formation, Bio-Informatics…") via the predicates (p1, p2, p3…). In its turn the resources "University, Formation, Bio-Informatics…" are linked via other predicates (Pj) with other resources (for example: "Boston University…"). Therefore, it appears relevant to recommend some courses of the Boston University to the current user.

Our approach is based on the iterative exploration of the DBpedia graph, where each step depends on the result of the previous steps.

In order to obtain relevant and personalized recommendations for each learner, we calculate the occurrence number of the {domain, author, year, edition…} characteristics and then we choose the ones that best reflect the learner interest to exploit them later in the exploration of the RDF graph of DBpedia.

The purpose of the graph exploration is to obtain recommendations that should not only satisfy the learner but also to have a diversity and a novelty in the recommendation, to create the effect of surprise by recommending resources that the learner did not expect them at the beginning. The learner evaluates the recommended resources in real time in each iteration. The process stops when none of the recommended resources was satisfied the user.

If the learner liked at least one resource among those in the proposed list, in the second iteration, we focus on these ones. Thus, we re-explore the LOD graph again starting from these items by using the query language sparql to return more educational resources connected with them; this technique allows us to propose a list of diverse and novel resources to ensure the surprise effect.

The real-time evaluation process as well as the exploration of the graph is iterative. At each iteration, we explore the graph based on the positive ratings assigned to the resources previously recommended. Indeed, the evaluation is an essential step to determine the new pattern of requests for the re-exploration of the graph to generate another list of recommendations. Every time, we propose to the user ten resources, if he assigns a rating more or equal to three, we consider that he liked the recommended resource, and so we record it in his profile, otherwise we move to another resource.

After evaluating the ten resources, the program suggests to the user to recommend even more ones. If he accepts then another list of resources is generated from his profile, otherwise, we stop and we return the list of resources liked. With this method, we ensure that the recommended list of resources is diverse, where every user can obtain diverse resources even they do not appear in the profile of his neighbors in the social network.

3.2 Using Ant Colony Optimization to Enhance Recommendation in LOD Graph

Until now, the major limitation is that in each iteration we will choose only the resources of the next level, this can limit the number of resources to recommend and also ignore completely the social aspect in the recommendation process (the user's neighbors is not be considered).

In order to remedy this problem, we suggest using the ACO algorithm to benefit from the strength of community aspect that characterizes the ants.

With the use of the ACO algorithm, we can recommend more resources to a user because we can easily explore the LOD graph without being limited to a specific level during the search, as well as the social aspect of our approach can be emerged without calculating the similarity between the users. This is done as follows:

Each user is represented by an ant. Each time when the user accepts the resource recommended by the system, the path is marked by a pheromone. And therefore the resources strongly accepted by the majority of users have more pheromone.

In this case, when we want to recommend resources to an active user, the system starts by seeing if the recommendation path is marked by a pheromone greater than or equal to a given threshold. If so, the system recommends all the resources of this path directly to this user.

Each resource Rx is defined by its features i.e. the triples of a kind (Rx, Predicate, Ry), where Predicate denote the type of the relation and Ry referred the target node (the node connected to the other end of the relation). In Fig. 2, we find two paths are marked by a pheromone: the first path R1-R2-R3-R4-R5 and the second one R7-R6-R5. We can conclude for example that the majority of users whom liked the resource R7, they liked also the resources R6 and R5.

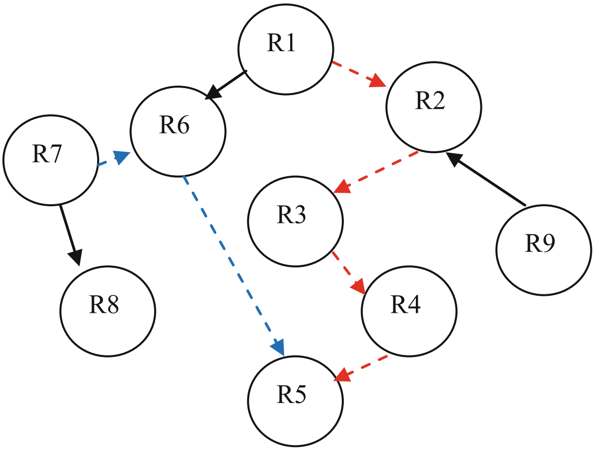


Fig. 2. A sub graph LOD with a pheromone

4 Experimental Results

In this section, experiment over a popular dataset is described and results are analyzed and discussed. The dataset exploited in our test is del.icio.us: a web-based social bookmarking tool which let a user manage a personal set of web resources and annotate them with tags. In this experiment, we were interested in data sample constructed from users who tagged resources about education. Thus, our database comprises 1712 tag assignments involving 150 users, 5430 tags, and 744 resources. We used DBpedia one of the most successful initiatives developed based on the Linked Open Data principles. In order to evaluate our LOD-based recommender system, we had to link resources in del.icio.us dataset to their corresponding resources in DBpedia.

4.1 Experimental Methodology

To evaluate the quality of a recommender system, we must demonstrate that the recommended resources are really being accepted and added by the users. Because the knowledge of this information requires asking the users of the selected databases if they appreciated the proposed set of resources, which is impossible in our case because we don't know this community, we have randomly removed some resources from the profile of each user, and we applied our approach on the remainder dataset in order to show if it can recommend the removed resources to their corresponding users or not. If it is the case, so we can conclude that our approach enables to extract the user preferences. To test the performance of our approach we proceed as follow:

(a) Evaluating the Precision, Recall and F1 Measure of Our Approach
In order to evaluate the quality of our recommender system, we have used the following three metrics: recall, precision, and F1 metric that is a combination of recall and precision. We calculated the three metrics in five iterations.

The curve presented in Fig. 3 shows the average of precision, recall and f1 measures in the five iterations. We notice that in the first iterations the precision achieved a good value equal to 0.77 this is due to the fact that the system recommends exactly the items wanted by the user in this case i.e. those that match his profile. At the end of the fifth iteration the system begins to deteriorate in terms of precision but always with a value that exceeds 0.76. This decrease in precision is quite normal since the system begins to recommend items according to different attributes (domain, year …) which is known as diversity of recommendation. Learners sometimes accept the recommended resources and other times it was not the case. Recall and F1 measure achieved all both good values in the all iterations.

(b) Diversity and Novelty in Recommendation
The goal of novelty and diversity in recommender systems is to reduce redundancy in the list of recommendations, and also to take into account the diverse interests of users and not just recommend the most popular or the most similar items. To calculate individual diversity and novelty, we used the metrics proposed in Zhang and Hurley (2009) and Vargas (2014) respectively. Figure 4 showed promising values of both diversity and novelty in the five iterations. This demonstrates the importance of Linked Open Data to extract more diversified and novel resources in the recommendation.

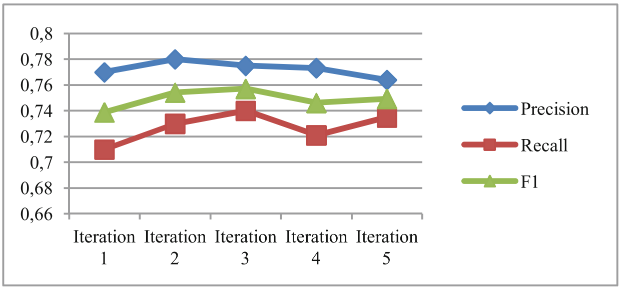


Fig. 3. Average precision, recall and F1 of the recommendations

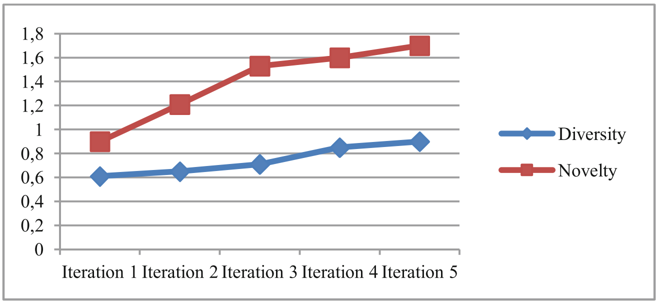


Fig. 4. Average of diversity and novelty

It is clear that the effectiveness of recommendation depends of preserving both precision and diversity. Figure 5 demonstrate that our approach preserving both them in all iterations.

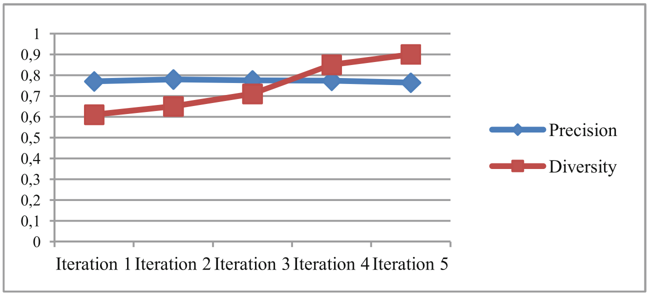


Fig. 5. Diversity vs. precision

4.2 Discussion

From the analysis of the above results, we can conclude that precision, recall and F1 metrics are very promising in del.icio.us dataset. These results indicate the force of using both LOD and ACO in recommender system. The system also ensured diversity and non-redundancy of the recommendations. Table 1 presents the deviation value of precision, recall, F1 metric, diversity and novelty.

Table 1. The standard deviation value of the three metrics

	Precision	Recall	F1	Diversity	Novelty
Standard deviation	5%	6%	5%	3.4%	7%

In all cases, these values are very small which indicates that the value of these measures for each user tend to be very close to the average. Since the averages (presented in Figs. 3 and 4) are very promising for the community in general, the small values of standard deviations indicate that the metrics are also promising for each learner individually.

4.3 Scale-Up Experiment

As recommender systems are designed to help users navigate in large collections of items, one of our goals is to scale up to real datasets. Therefore, it is important to measure how fast does our approach provides recommendations. In this subsection, we discuss the impact of increasing the number of learners on the execution time of our approach. In order to demonstrate the scalability of our approach, we measured the execution time required to make relevant recommendations in del.icio.us database, with a number of users increasing from 20 to 150 users. Figure 6 shows that the execution time (in seconds) of our approach linearly increases as the database size increase, meaning that our approach have relatively good scale-up behavior since the increase of the number of learners in the database will lead to approximately the linear growth of the processing time, which is desirable in the processing of large databases.

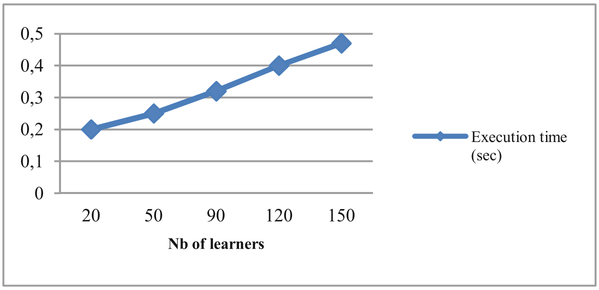


Fig. 6. Performance of our approach when the database size increases

5 Conclusion

Our investigations in the field of E-learning folksonomies have allowed us to make a contribution in which we are interested to personalize resources retrieval according to learners' interest. In this contribution we have exploited the strength of social aspect in folksonomies to let members in the community benefit from the educational resources tagged by the others one based on resources recommendation. The proposed approach is based on LOD exploration and ACO algorithm. The objective was overcoming the problem of diversity and novelty in recommendation. Primary results show also the utility of exploring LOD graph in ensuring diversity when recommending personalized educational resources in social tagging systems. We have tested the approach on a baseline dataset and we have obtained promising results. In order to continue and improve our work, we aim at using others principles like event detection, for example, to help capturing and analyzing the behavior of learners when new events come, this can improve recommendation and even resources ranking. Also, we intend to test our approach with other algorithms like the genetic algorithm.

References

Beldjoudi, S., Seridi, H., Bnzine, A.: The impact of social similarities and event detection on ranking retrieved resources in collaborative e-learning systems. In: Koch, F., Koster, A., Primo, T., Guttmann, C. (eds.) CARE/SOCIALEDU-2016. CCIS, vol. 677, pp. 47–63. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-52039-1_4

Beldjoudi, S., Seridi, H., Faron-Zucker, C.: Personalizing and improving resource recommendation by analyzing users preferences in social tagging activities. Comput. Inform. **36**(1), 223–256 (2017)

Berners-Lee, T.: (2007). http://www.w3.org/DesignIssues/LinkedData.html

Lau, S.B.-Y., Lee, C., Singh, Y.P.: A folksonomy-based lightweight resource annotation metadata schema for personalized hypermedia learning resource delivery. Interact. Learn. Environ. **23**(1), 79–105 (2015)

Karabadji, N.E.I., Beldjoudi, S., Seridi, H., Aridhi, S., Dhifli, W.: Improving memory-based user collaborative filtering with evolutionary multi-objective optimization. Expert Syst. Appl. **98**, 153–165 (2018)

Klašnja-Milićević, A., Ivanović, M., Nanopoulos, A.: Artif. Intell. Rev. **44**, 571 (2015). https://doi.org/10.1007/s10462-015-9440-z

Kopeinik, S., Lex, E., Seitlinger, P., Albert, D., Ley, T.: Supporting collaborative learning with tag recommendations: a real-world study in an inquiry-based classroom project. In: Proceedings of 7th International Learning Analytics & Knowledge Conference, LAK 2017, pp. 409–418 (2017)

Torniai, C., Jovanovic, J., Bateman, S., Gaševic, D., Hatala, M.: Leveraging folksonomies for ontology evolution in e-learning environments. In: 2008 IEEE International Conference on Semantic Computing, pp. 206–213 (2008)

Vargas, S.: Novelty and diversity enhancement and evaluation in recommender systems and information retrieval. In: Proceedings of 37th International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2014, New York, NY, USA, p. 1281, (2014)

Zhang, M., Hurley, N.: Novel item recommendation by user profile partitioning. In: 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology, vol. 1, pp. 508–515, September 2009

Evaluating Adaptive Pedagogical Agents' Prompting Strategies Effect on Students' Emotions

François Bouchet[1](✉), Jason M. Harley[2], and Roger Azevedo[3]

[1] Laboratoire d'Informatique de Paris 6, LIP6, CNRS, Sorbonne Université, 75005 Paris, France
francois.bouchet@lip6.fr

[2] Educational Psychology, University of Alberta, Edmonton, AB, Canada
jharley1@ualberta.ca

[3] Psychology, North Carolina State University, Raleigh, NC, USA
razeved@ncsu.edu

Abstract. Adapting ITSs that promote the use of metacognitive strategies can sometimes lead to intense prompting, at least initially, to the point that there is a risk of it feeling counterproductive. In this paper, we examine the impact of different prompting strategies on self-reported agent-directed emotions in an ITS that scaffolds students' use of self-regulated learning (SRL) strategies, taking into account students' prior knowledge. Results indicate that more intense initial prompting can indeed lead to increased frustration, and sometimes boredom even toward pedagogical agents that are perceived as competent. When considering prior knowledge, results also show that this strategy induces a significantly different higher level of confusion in low prior knowledge students when compared to high prior knowledge students. This result is consistent with the fact that higher prior knowledge students tend to be better at self-regulating their learning, and it could also indicate that some low prior knowledge students may be on their path to a better understanding of the value of SRL.

Keywords: Adaptivity · Prompting · Pedagogical agents
Intelligent tutoring systems · Emotions · Affects · Metacognition
Self-regulated learning

1 Introduction

Adaptation is key to successful learning with intelligent tutoring systems (ITSs), as learners integrate instructional material with elements ITSs are designed to enhance: learning and problem solving [1]. But an efficient short-term teaching strategy can backfire if the learner starts experiencing negative emotions, including those directed toward the tutor. Many studies have shown the benefit of metacognitive prompting to encourage learners to deploy self-regulated learning (SRL) strategies [2]. The idea is to move from co-regulated learning [3], with the assistance of a human tutor or of a pedagogical agent, toward real self-regulation, with the assumption that these skills will transfer beyond the learning session. Because of the relatively short time of interaction with the ITS, one can be tempted to initially choose a high-frequency prompting strategy

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 33–43, 2018.
https://doi.org/10.1007/978-3-319-91464-0_4

which progressively decreases as the learner engages in monitoring their learning and deploying learning strategy on their own—a strategy which has been shown to be beneficial for the use of SRL processes, without reducing the perceived usefulness of the ITS [4]. However, the more insistent or more frequent the interventions, the more likely they may be to trigger emotional reactions from learners. This is important because emotions are critical to fostering motivation and the use of SRL strategies [5]. Therefore, there is a potential risk that eliciting negative emotions through intensive SRL (e.g., cognitive strategies, metacognitive monitoring) process prompting could eventually lead to a form of non-compliance or contempt toward the tutor, and potentially minimize the benefits of SRL. Moreover, as learners with high prior knowledge deploy different SRL strategies from learners with low prior knowledge [6], it is likely their reactions to adaptive prompting could differ.

In this study, we investigated agent-directed emotions self-reported by learners after their interaction with an open-ended learning environment embedded with several pedagogical agents (PAs) that promoted different facets of SRL. We focused particularly on negatively-valenced emotions because of the deleterious impact they can have on learning (in our case, the benefits of SRL) – although exceptions exist such as confusion, which sometimes is positively associated with deep learning [7]. Specifically, we examined three separate but related research questions: (RQ1) Did the frequency of prompts from different PAs affect emotions directed toward them? (RQ2) How did students report feeling toward PAs in two different prompting rule conditions (an initially high but decreasing prompting strategy and a lower intensity prompting strategy)? And (RQ3) whether high prior knowledge students' emotional reactions to adaptive prompting differed from low prior knowledge students? We hypothesized that the higher the number of prompts from an agent, the more likely a learner would be to experience negatively-valenced emotions. We also hypothesized that students with higher initial prompting would be likely to experience more negatively-valanced emotions, and that higher prior knowledge learners would also experience more negative emotions from being told what they may already know with more prompts to self-regulate. We did not investigate the effect of the adaptive prompting on learning outcomes because we have already shown its neutral to mildly positive effects in a previous work [4].

The emotions learners experience while interacting with ITSs is a widely studied topic, and one examined using a variety of methodologies. As with traditional psychological studies, self-report measures are popular instruments for measuring emotions. The Academic Emotions Questionnaire (AEQ) [8] is one such measure; it has been used in research with MetaTutor (but not considered here) and BioWorld [9, 10]. The Online Motivation Questionnaire (OMQ) [11] has also been used in experiments with iSTART [12] or BioWorld [13] to measure emotions. In addition to self-report measures, ITS researchers have also used various behavioral and physiological approaches, including facial expression recognition (e.g. CERT [14]), electrodermal activity [15], and body language [16]. Online and multimodal approaches to emotion measurement, provide many advantages over self-report, but were not as relevant for our research questions here because we are particularly interested in emotions directed towards a specific part of the ITS: the pedagogical agents. As such, it was most appropriate to use a self-report measure to focus students' answers.

2 Method

2.1 Participants and Experimental Conditions

One hundred and sixteen undergraduate students ($N = 116$, 17–31 years old, $M = 20.9$ years, $SD = 2.4$; 64.6% female; 62.9% Caucasian) from two North American Universities, studying different majors and with various levels of prior knowledge participated in this study. Each participant received $50 upon completion of the study and was randomly assigned to one of three conditions: (1) *non-adaptive prompt* (NP – $n = 58$), (2) *frequency-based adaptive prompt* (FP – $n = 29$) and (3) *frequency and quality-based adaptive prompt* (FQP – $n = 29$). Participants from the adaptive conditions, FP and FQP, were grouped in some analyses, leading to two samples of identical sizes.

Non-adaptive Prompt (NP). In the NP condition, learners received a moderate but constant amount of prompts from the PAs (on average, 1 per 10 min) to engage in various SRL processes throughout the learning session. Previous PA prompts, learners' initiative to enact SRL processes, validity of learners' metacognitive judgments or efficiency at using a learning strategy had no impact on the prompts from the PAs.

Frequency-Based Adaptive Prompt (FP). In the FP condition, learners received more prompts at the beginning of the session (on average, 3.5 per 10 min), but the probability of both categories of prompts (monitoring and strategy) being triggered decreased after each new prompt was received and after each self-initiated enactment of an SRL process by the learner. Accordingly, participants who had been prompted frequently at first *and* who had been self-initiating SRL processes regularly could potentially end up receiving no further prompts by the end of the session.

Frequency and Quality-Based Adaptive (FQP). The FQP condition applies the same prompt deceasing rules as the FP condition with the addition of two further rules that (if triggered) will *increase* the probability of monitoring or learning strategy prompts of being triggered. If (1) the learner does not comply with a PA's (non-mandatory) prompt

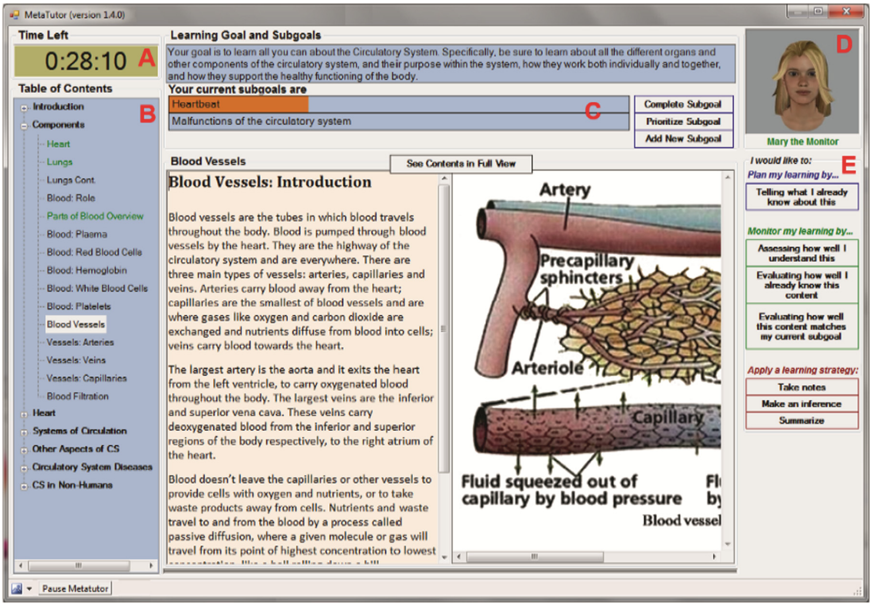
Table 1. Condition of successes associated to the different type of SRL prompts.

Type	Type of PA's prompt	Condition of success
Monitoring	Judgment of Learning (JOL)	Accurate evaluation of what has been learnt
	Feeling of Knowing (FOK)	Accurate evaluation of what is already known
	Content Evaluation (CE)	Accurate evaluation of the relevance of the content relative to the active sub-goal
	Management of Progress Toward Goal (MPTG)	Learner validates their sub-goal in the next 45 s
Strategy	Summarization (SUMM)	If learner delays, must be performed later on
	Coordination of Information Sources (COIS)	Image is opened in the next 45 s
	Draw image already opened	Digital notepad in the next 45 s
	Draw image not opened yet	Learner accepts to open the image

to deploy a certain learning strategy (i.e., to re-read a page), or (2) a learner's metacognitive judgment was inaccurate (e.g., selected a page as relevant to his/her active learning when it was not; cf. Table 1 for the list of conditions of success) then the probability of both categories of prompts being triggered will increase.

2.2 The Testbed System, Experimental Procedure and Data Used

System Overview. MetaTutor is an intelligent, hypermedia learning environment in which four embedded PAs help the student to learn more efficiently by prompting them to engage in SRL processes (*cf.* Fig. 1). They navigate through the 38 pages (with text and images) on human circulatory system using a table of contents (noted B in Fig. 1). Progress toward the overall learning goal and the sub-goals chosen at the beginning of the session is always visible at the top of the system interface (C in Fig. 1). A timer displays the time remaining in the learning session (A in Fig. 1). One of the four PAs is always visible in the top right-hand corner of the interface (D in Fig. 1), corresponding to the last one who interacted with the student (using text and voice as output, but text-only as input for students' answers to the prompts). The PAs' appearances and voices are the same in each experimental condition, and each PA is comparable in terms of visual and audio quality. Each PA has a specific role: *Pam the Planner* helps the student to plan their learning sub-goals, *Mary the Monitor* helps in monitoring the learning, *Sam the Strategizer* assists with the deployment of learning strategies and *Gavin the Guide* introduces the system and its questionnaires. PAs' prompts are triggered depending on parameters such as the time spent on a page or the relevance of the page to students'



Your goal is to learn all you can about the Circulatory System. Specifically, be sure to learn about all the different organs and other components of the circulatory system, and their purpose within the system, how they work both individually and together, and how they support the healthy functioning of the body.

Blood Vessels: Introduction

Blood vessels are the tubes in which blood travels throughout the body. Blood is pumped through blood vessels by the heart. They are the highway of the circulatory system and are everywhere. There are three main types of vessels: arteries, capillaries and veins. Arteries carry blood away from the heart; capillaries are the smallest of blood vessels and are where gases like oxygen and carbon dioxide are exchanged and nutrients diffuse from blood into cells; veins carry blood towards the heart.

The largest artery is the aorta and it exits the heart from the left ventricle, to carry oxygenated blood throughout the body. The largest veins are the inferior and superior vena cava. These veins carry deoxygenated blood from the inferior and superior regions of the body respectively, to the right atrium of the heart.

Blood doesn't leave the capillaries or other vessels to provide cells with oxygen and nutrients, or to take waste products away from cells. Nutrients and waste travel to and from the blood by a process called passive diffusion, where a given molecule or gas will travel from its point of highest concentration to lowest

Fig. 1. Annotated screenshot of the system interface.

current sub-goal. Additional parameters allow to adjust the triggering to obtain an overall higher/lower frequency of prompts (conditions FP and FQP) and to consider compliance and accuracy of previous SRL processes (condition FQP). Below the PA, a palette of buttons allows students to self-initiate SRL processes, leading to a sequence of steps very similar to when the prompt comes from a PA: an invitation to perform the process followed by a feedback on its validity (e.g. agreeing the page is relevant to the current learning sub-goal).

Experimental Procedure. Participants used the system individually on a desktop computer in two sessions separated by one hour to three days. During session 1 (30 to 40 min. long), they filled and signed a consent form and completed several computer-based self-report questionnaires, a demographics survey and a 25-item pre-test on the circulatory system. During session 2 (90 min. long), participants used MetaTutor to learn about the circulatory system. Participants had 60 min to interact with the content during which they could initiate SRL processes or do so after a PA's prompt. MetaTutor was paused when participants were watching a video, taking a survey, and during an optional 5 min break half-way through the session. At the end of the session, participants were given a post-test and filled a questionnaire, the Agent Response Inventory (ARI) [17], which included statements on the emotions each agent made them feel (e.g. "SAM made me feel frustrated") that they had to rate on a 5-point Likert scale (from "strongly disagree" to "strongly agree").

Data Coding and Scoring. Because only prompts from Mary and Sam varied between conditions, we focused on emotions toward these two agents. 19 emotions were assessed, but we focused on the negatively-valenced ones (the most deleterious on learning, as mentioned before). When two emotions were very close from each other (e.g. anger/ frustration, fear/anxiety, disgust/contempt) we also chose to remove one of the two, on the basis that non-expert students could fail to grasp the real but subtle nuance that exists. In each case, we kept the emotion in the pair that seemed to be the more learning-oriented (e.g. frustration over anger, anxiety over fear) or social (contempt over disgust) one. We ended up with a set of 7 emotions: frustration, anxiety, shame, hopelessness, boredom, contempt, and confusion.

To evaluate the frequency of prompts received by each participant in each condition, we extracted from log-file data the average number of prompts they received from each PA over a period of 10 min. Finally, to determine prior knowledge level, we used the adjusted ratio (between 0 and 1) of correct answers in the pre-test[1]. We conducted a median-split on participants' adjusted pretest score, such that participants whose scores fell below the median were labeled as low prior knowledge (LPK) and those who scored above were labeled as high prior knowledge (HPK). 2 participants whose score was equal to the median value (0.727) were excluded. Scores in the LPK group (*n*been prompted frequently 57) varied from 0.125 to 0.722 ($M = 0.523$ and $SD = 0.149$) and scores in the HPK group ($n = 57$) varied from 0.733 to 1 ($M = 0.891$, $SD = 0.072$).

[1] We selected only items among the 25 questions that were relative to the subgoals each participant set at the beginning of their learning session (as participants did not have time to explore all the learning material relative to each of the 7 subgoals available).

3 Results

3.1 Effect of Agents' Prompts Frequency on Agent-Directed Emotions

Pearson product-moment correlations were run to determine the relationship between number of prompts per period of 10 min from Mary/Sam and the score of each emotion toward Mary/Sam. There were significant positive correlations between (a) number of prompts and frustration toward Mary (r = .238, p = .010), (b) frustration toward Sam (r = .338, p = .000), and (c) boredom toward Mary (r = 0.190, p = .041). Other emotions were not statistically significantly correlated.

3.2 Effect of Adaptive Prompting on Agent-Directed Emotions

Mann-Whitney tests were run to examine differences between learners in conditions NP and FP&FQP in terms of emotions toward Mary/Sam. The results indicated that frustration was higher toward Mary (U = 1155.5, p = .001) in condition FP&FQP (M = 2.83) than in NP (M = 2.10). Regarding Sam, results indicated that frustration was higher (U = 1274, p = .010) in condition FP&FQP (M = 2.66) than in NP (M = 3.31), and that shame was marginally lower (U = 1886.5, p = .091) in FP&FQP (M = 1.47) than in NP (M = 1.81). No statistically significant results were found for the other emotions (*cf.* Table 2).

Table 2. Average self-reported emotions towards Sam and Mary in both conditions

Condition	NP				FP&FQP			
Agent	Sam		Mary		Sam		Mary	
Emotion	*M*	*SD*	*M*	*SD*	*M*	*SD*	*M*	*SD*
Frustration	2.66^{**}	1.57	2.10^{***}	1.19	3.31^{**}	1.43	2.85^{***}	1.42
Anxiety	2.26	1.40	2.21	1.28	2.17	1.32	2.16	1.32
Shame	1.81^{*}	1.21	1.62	0.93	1.47^{*}	0.90	1.67	1.14
Hopelessness	1.69	1.12	1.55	0.89	1.51	1.02	1.57	1.07
Boredom	2.38	1.22	2.12	1.16	2.53	1.45	2.28	1.34
Contempt	2.31	1.47	1.90	1.13	1.98	1.35	1.91	1.36
Confusion	2.02	1.42	1.60	0.93	1.90	1.31	1.79	1.24

$^{*}p < 0.10$; $^{**}p < 0.05$; $^{***}p < 0.01$

3.3 Effect of Prior Knowledge on Agent-Directed Emotions

First, we examined the existence of an interaction between prior knowledge (groups LPK and HPK) and conditions. We ran multiple two-way ANOVAs to examine the effect of prior knowledge and condition on emotions toward Mary and Sam. No statistically significant interaction between prior knowledge and condition was revealed, leading us to consider prior knowledge individually.

Then as in Sect. 3.2, we ran two sets of Mann-Whitney tests to examine differences between HPK and LPK learners in (1) condition NP (*cf.* Table 3), and condition (2; merged)

FP&FQP (*cf.* Table 4). In condition NP, no statistically significant results were found between LPK and HPK learners for the 7 emotions tested. In condition FP&FQP, however, the tests revealed that hopelessness was higher toward Sam ($U = 499$, $p = .009$) for LPK ($M = 1.92$) than for HPK learners ($M = 1.22$). It was also the case with higher confusion ($U = 501$, $p = .016$) for LPK ($M = 2.38$) than for HPK ($M = 1.56$). Conversely, HPK learners reported marginally more contempt ($U = 298.5$, $p = .064$) and frustration ($U = 289$, $p = .051$) towards Sam. Similar patterns were found for Mary who elicited more confusion ($U = 494$, $p = .021$) for LPK ($M = 2.08$) than for HPK ($M = 1.59$), and marginally more hopelessness for LPK ($U = 448$, $p = .097$).

Table 3. Average self-reported emotions towards Sam and Mary in condition NP

Condition	Low prior knowledge				High prior knowledge			
Agent	Sam		Mary		Sam		Mary	
Emotion	*M*	*SD*	*M*	*SD*	*M*	*SD*	*M*	*SD*
Frustration	2.49	1.54	2.06	1.25	2.88	1.58	2.16	1.08
Anxiety	2.27	1.44	2.09	1.31	2.24	1.34	2.36	1.23
Shame	1.79	1.20	1.58	0.89	1.84	1.22	1.68	0.97
Hopelessness	1.73	1.14	1.63	0.98	1.64	1.09	1.44	0.75
Boredom	2.24	1.35	2.27	1.36	2.56	0.98	1.92	0.80
Contempt	2.36	1.47	1.94	1.13	2.24	1.45	1.84	1.12
Confusion	1.91	1.42	1.61	0.95	2.16	1.41	1.60	0.89

Table 4. Average self-reported emotions towards Sam and Mary in condition FP&FQP

Condition	Low prior knowledge				High prior knowledge			
Agent	Sam		Mary		Sam		Mary	
Emotion	*M*	*SD*	*M*	*SD*	*M*	*SD*	*M*	*SD*
Frustration	3.00^{*}	1.47	2.83	1.31	3.63^{*}	1.34	2.84	1.52
Anxiety	2.08	1.36	2.00	1.19	2.19	1.29	2.22	1.41
Shame	1.63	0.95	1.67	1.07	1.38	0.86	1.69	1.21
Hopelessness	1.92^{***}	1.19	1.71^{*}	1.02	1.22^{***}	0.78	1.41^{*}	1.09
Boredom	2.38	1.32	2.25	1.20	2.63	1.56	2.28	1.46
Contempt	1.71^{*}	1.10	2.00	1.29	2.22^{*}	1.49	1.88	1.43
Confusion	2.38^{**}	1.41	2.08^{**}	1.12	1.56^{**}	1.14	1.59^{**}	1.32

$^{*}p < 0.10$; $^{**}p < 0.05$; $^{***}p < 0.01$

4 Discussion

The first two results confirm our initial hypothesis: both agents elicited more negative emotions when more prompts were received, even if their frequency had decreased by the end of the learning session (for most participants, it was below the frequency of prompts received in condition NP). We can assume that the frustration associated with both agents was mostly related to the increased disruptions in the learning task, which was probably stronger initially (in conditions FP&FQP) and hadn't decayed by the end. This could be an issue because frustration can be a useful emotion when temporary and directed toward the learning material, but not necessarily if directed toward a tutor. Regarding other negative emotions, the fact that they are different between Mary (whose additional prompting also was accompanied by increased boredom) and Sam (whose additional prompting led to lower level of shame) indicates that the differences in emotions stems from the agents' roles (cf. Sect. 2.2). Overall, Mary's feedback is more immediately helpful to the student, even when it is negative (e.g. "this page is actually not relevant", "you don't seem to know this content as well as you thought"). Repetitiveness of such feedback simply reveals the limits of what the PA can provide. On the contrary, Sam's feedback can be perceived as more judgmental ("your summary was a little long/short") without necessarily being immediately helpful. Repetitiveness leads to an inhibition of the initial shame learners can have when failing to deploy the suggested strategies—although reduced shame would be positive for learning, the conjunction with increased frustration makes us assume that learners were probably just becoming more dismissive of Sam's feedback.

The third result goes against our initial hypothesis: not only did high prior knowledge participants not feel more frustration toward the agents, but when there was a difference with low prior knowledge students, they reported feeling less negatively-valenced emotions. The fact that no differences existed in condition NP also means that the differences between LPK and HPK students appeared because of the more intense prompting initially (but was not directly related to the total amount of prompts, as shown by the first analysis). We already know that HPK learners deploy their SRL strategies differently from LPK ones [6], and that HPK learners tend to naturally use more the system prompts to regulate their learning [18]. Therefore, the additional confusion for LPK learners can either mean (a) that they did not perceive the point of agents' prompts (or of SRL altogether), and that the increased intensity only made them wonder more about their interest, or (b) that the confusion was only felt initially, which can be a desirable initial state for learning, and that later on in the session, they were starting to perceive the value of agents' prompts. The fact that the system usefulness was not perceived lower in conditions FP&FQP tends to indicate that at least some LPK learners were in that situation [4]. This is an encouraging result, as it shows that despite the limits of the PAs' range of prompts and the repetitiveness of some of their feedback, some low prior knowledge students (who are the ones who can benefit the most from self-regulating their learning) managed to perceive more the value of the self-regulation fostered by the system.

5 Conclusion, Limits and Future Works

Overall, this study shows that adaptive prompting with a more intense initial strategy is a double-edged sword: on the one hand, it emphasizes the limits of the ITS and its embedded PAs, whose smallest flaws become magnified and prone to increased frustration. On the other hand, although high prior knowledge participants quickly seem to understand the benefits of PAs' help (even if it is of low intensity), having more frequent prompts seems to help low prior knowledge participants more, after what we can assume to be a temporary initial confusion. This help is limited, however, by the usefulness of the prompts, which we have previously seen, may not be specific enough to lead to a significantly measurable increase in the learning outcome (when comparing pre to post-test results) [4].

One of the limits of this study is that we only measured participants' feelings toward each agent for the overall session at its end (and at one point in time). We therefore cannot, for example, rule out that LPK students finished their learning session more confused than in the non-adaptive prompting condition. Using constant emotion monitoring, for instance, through automatic facial analysis, could help with this issue [19]. It would also decrease the reliance on self-report, as students can sometimes be poor reporters of their own emotional state. It is worth noting, however, that even if we have previously found good agreement between automatic facial expression analysis and self-report; facial expression recognition software has typically focused on basic emotions [15], to the exclusion of some achievement-related emotions such as boredom and confusion. Another limit is the fact that we considered conditions FP and FQP together in all analyses on account of sample size. Evaluating these two adaptive conditions separately represents an important future direction Finally, we lack long-term evaluations of whether participants have indeed internalized more the benefit of using externally-prompted, and sometimes collaborative [20], self-regulated learning strategies, which is the goal of a system such as MetaTutor. Using information from students' emotional responses toward MetaTutor and its PAs to provide real-time, user-adaptive [21] SRL prompts also represents an important future direction for this work.

Acknowledgements. Research supported by funding from NSF (DRL 1008282, DRL1431552, DRL 1660878), SSHRC, and CRC program awarded to third author.

References

1. Ma, W., Adesope, O.O., Nesbit, J.C., Liu, Q.: Intelligent tutoring systems and learning outcomes: a meta-analysis. J. Educ. Psychol. **106**, 901–918 (2014)
2. Bannert, M., Mengelkamp, C.: Scaffolding hypermedia learning through metacognitive prompts. In: Azevedo, R., Aleven, V. (eds.) International Handbook of Metacognition and Learning Technologies, vol. 28, pp. 171–186. Springer, New York (2013). https://doi.org/10.1007/978-1-4419-5546-3_12
3. Järvelä, S., Hadwin, A.F.: New frontiers: regulating learning in CSCL. Educ. Psychol. **48**, 25–39 (2013)
4. Bouchet, F., Harley, J.M., Azevedo, R.: Can adaptive pedagogical agents' prompting strategies improve students' learning and self-regulation? In: Micarelli, A., Stamper, J., Panourgia, K. (eds.) ITS 2016. LNCS, vol. 9684, pp. 368–374. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39583-8_43
5. Pekrun, R., Perry, R.P.: Control-value theory of achievement emotions. In: Pekrun, R., Linnenbrink-Garcia, L. (eds.) International Handbook of Emotions in Education, pp. 120–141. Routledge, Abingdon (2014)

6. Taub, M., Azevedo, R., Bouchet, F., Khosravifar, B.: Can the use of cognitive and metacognitive self-regulated learning strategies be predicted by learners' levels of prior knowledge in hypermedia-learning environments? Comput. Hum. Behav. **39**, 356–367 (2014)
7. D'Mello, S., Lehman, B., Pekrun, R., Graesser, A.: Confusion can be beneficial for learning. Learn. Instr. **29**, 153–170 (2014)
8. 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. Educ. Psychol. **37**, 91–105 (2002)
9. Harley, J.M., Bouchet, F., Azevedo, R.: Examining how students' typical studying emotions relate to those experienced while studying with an ITS. In: 14th International Conference on Intelligent Tutoring Systems. Springer International Publishing, Montreal (2018)
10. Jarrell, A., Harley, J.M., Lajoie, S., Naismith, L.: Success, failure and emotions: examining the relationship between performance feedback and emotions in diagnostic reasoning. Educ. Technol. Res. Dev. **65**, 1263–1284 (2017)
11. Boekaerts, M.: The on-line motivation questionnaire: a self-report instrument to assess students' context sensitivity. In: Pintrich, P.R., Maehr, M.M. (eds.) New Directions in Measures and Methods, pp. 77–120. Emerald Group Publishing Limited, Bingley (2002)
12. Jacovina, M.E., Tanner Jackson, G., Snow, E.L., McNamara, D.S.: Timing game-based practice in a reading comprehension strategy tutor. In: Micarelli, A., Stamper, J., Panourgia, K. (eds.) ITS 2016. LNCS, vol. 9684, pp. 59–68. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39583-8_6
13. Lajoie, S.P., Naismith, L., Poitras, E., Hong, Y.-J., Cruz-Panesso, I., Ranellucci, J., Mamane, S., Wiseman, J.: Technology-rich tools to support self-regulated learning and performance in medicine. In: Azevedo, R., Aleven, V. (eds.) International Handbook of Metacognition and Learning Technologies, pp. 229–242. Springer, New York, New York, NY (2013). https://doi.org/10.1007/978-1-4419-5546-3_16
14. Littlewort, G., Whitehill, J., Wu, T., Fasel, I., Frank, M., Movellan, J., Bartlett, M.: The computer expression recognition toolbox (CERT). Face Gesture **2011**, 298–305 (2011)
15. Harley, J.M., Bouchet, F., Hussain, M.S., Azevedo, R., Calvo, R.A.: A multi-componential analysis of emotions during complex learning with an intelligent multi-agent system. Comput. Hum. Behav. **48**, 615–625 (2015)
16. D'Mello, S.K., Graesser, A.: Multimodal semi-automated affect detection from conversational cues, gross body language, and facial features. User Model. User-Adapt. Interact. **20**, 147–187 (2010)
17. Harley, J.M., Carter, C.K., Papaioannou, N., Bouchet, F., Landis, R.S., Azevedo, R., Karabachian, L.: Examining the predictive relationship between personality and emotion traits and students' agent-directed emotions: towards emotionally-adaptive agent-based learning environments. User Model. User-Adapt. Interact. **26**, 177–219 (2016)
18. Bouchet, F., Kinnebrew, J.S., Biswas, G., Azevedo, R.: Identifying students' characteristic learning behaviors in an intelligent tutoring system fostering self-regulated learning. In: Yacef, K., Zaïane, O., Hershkovitz, A., Yudelson, M., Stamper, J. (eds.) Proceedings of 5th International Conference on Educational Data Mining, Chania, Greece, pp. 65–72 (2012)
19. Mudrick, N., Rowe, J.P., Taub, M., Lester, J.C., Azevedo, R.: Toward affect-sensitive virtual human tutors: the influence of facial expressions on learning and emotion. In: Busso, C., Epps, J. (eds.) Proceedings of 2017 7th International Conference on Affective Computing and Intelligent Interaction (ACII), pp. 184–189. IEEE Computer Society, Washington, DC (2017)

20. Harley, J.M., Taub, M., Azevedo, R., Bouchet, F.: "Let's set up some subgoals": understanding human-pedagogical agent collaborations and their implications for learning and prompt and feedback compliance. IEEE Trans. Learn. Technol. **11**, 54–66 (2018)
21. Harley, J.M., Lajoie, S.P., Frasson, C., Hall, N.C.: Developing emotion-aware, advanced learning technologies: a taxonomy of approaches and features. Int. J. Artif. Intell. Educ. **27**, 268–297 (2017)

Investigating the Role of Goal Orientation: Metacognitive and Cognitive Strategy Use and Learning with Intelligent Tutoring Systems

Elizabeth B. Cloude(✉), Michelle Taub, and Roger Azevedo

Laboratory for the Study of Metacognition and Advanced Learning Technologies,
Department of Psychology, North Carolina State University, Raleigh, NC, USA
{ebcloude,mtaub,razeved}@ncsu.edu

Abstract. Cognitive, affective, metacognitive, and motivational (CAMM) processes are critical components of self-regulated learning (SRL) essential for learning and problem solving. Currently, ITSs are designed to foster cognitive, affective, and metacognitive (CAM) strategies and processes, presenting major gaps in the research since motivation is a key component of SRL and influences the remaining CAM processes. In our study, students interacted with MetaTutor, a hypermedia-based ITS, to investigate how 190 undergraduate students' proportional learning gain (PLG) related to sub-goals set, cognitive strategy use and metacognitive processes differed based on self-reported achievement goal orientation. Results indicated differences between approach, avoidance, and students who adopted both approach and avoidance goal orientations, but no differences between mastery, performance and students who adopted both mastery and performance goal orientations on PLG for content related to sub-goal 1. Conversely, no differences were found between goal orientation groups on PLG for sub-goal 2, revealing possible changes in goal orientation following sub-goal 1. Analyses indicated no differences between goal orientation groups on metacognitive processes and cognitive strategy use. Thus, we suggest turning away from self-report data, where future studies aim to incorporate multi-channel data over durations of tasks as students interact with ITSs to measure motivation and its tendency to fluctuate in real-time. Implications for using multiple data channels to measure motivation could contribute to adaptive ITS design based on all CAMM processes.

Keywords: Achievement goal orientation · Motivation
Intelligent tutoring systems

1 Introduction

Intelligent tutoring systems (ITSs) have been designed to train students to effectively deploy self-regulated learning (SRL) strategies and processes such as monitoring and regulation that contribute to improved academic performance [1]. SRL is defined as planning and strategizing self-initiated actions of monitoring and adaptation of thoughts and behaviors to meet contextual demands and goals [2]. Four critical processes are

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 44–53, 2018.
https://doi.org/10.1007/978-3-319-91464-0_5

involved in SRL: cognitive, affective, metacognitive, and motivational (CAMM) [1]. Currently, ITS design only incorporates CAM (i.e., cognitive, affective, metacognitive) processes [3], presenting a major challenge in ITS development since motivation is a key aspect of SRL and influences the other CAM processes, and where without it, learning would be nonexistent [4].

Studies have found differences between motivated and unmotivated learners, specifically focusing on academic performance [4] and achievement goal orientation, a construct of motivation that explains goal-directed behaviors and offers insight into motivational states. There are two dimensions of goal orientation, with each dimension containing two components: mastery and performance and approach and avoidance. The mastery and performance dimension focuses on competence, where mastery-based students view competence as a product of effort and performance-based students perceive competence as outperforming peers. The approach and avoidance dimension focuses on valence, where approach-based students view competence with a positive attitude (e.g., opportunity to succeed) and avoidance-based students perceive competence in a negative light (e.g., possibility of failure) [5]. Research has heavily studied the mastery and performance dimension and revealed differences between mastery- and performance-based students on learning and affecting SRL strategy use [5, 6] where mastery-oriented students deploy more strategies relative to performance-oriented students. In contrast, research indicated mixed results when considering approach and avoidance and self-regulated strategy use [7].

Gaps in literature exist where researchers are relying on self-report questionnaires to measure achievement goal orientation in ITSs at one-time point. Researchers administer self-report measures *once* before students interact with an ITS as opposed to *multiple times* over a learning session as students interact with an ITS to see if goal orientation changes. In addition, the Achievement Goal Questionnaire-Revised (AGQ-R) classifies individuals into one component of each dimension (e.g., mastery-approach) and our data suggest not all individuals are that easily classified: some students scored the same on each component of both dimensions for goal orientation. To counter this issue, we created a new classification of achievement goal orientation that describes students who adopt both components of each dimension. For instance, students who scored the same on mastery and performance were assigned to a combination group for mastery and performance and students who scored the same on approach and avoidance were assigned to a combination group for approach and avoidance.

Thus, the current study aims to investigate whether goal orientation classifications based on AGQ-R scores used in MetaTutor, an ITS that allows students to set multiple sub-goals and provides tools for SRL strategy use, are consistent with literature where differences in goal orientation are indicative of differences in learning and SRL strategy use. More specifically, we aim to assess if there are differences in proportional learning gain (PLG), based on different sub-goals set, and cognitive and metacognitive processes between goal orientation groups. Based on previous literature, we hypothesize there will be differences in PLG based on different sub-goals, cognitive strategy use, and metacognitive processes between these groups [4, 6].

2 Methods

2.1 Participants and Measures

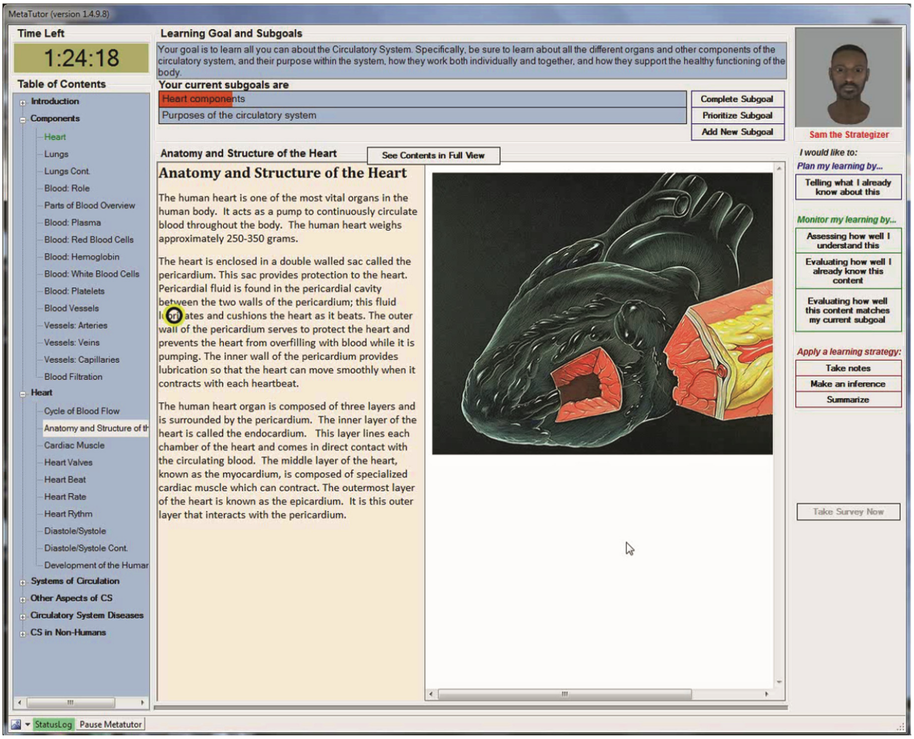
190 undergraduate students (52% female) recruited at three large North American universities completed a 2-day study with MetaTutor, an ITS aimed at teaching students about the circulatory system. Age ranged from 18 to 41 ($M = 20.43$, $SD = 2.94$) and students were compensated $10/hr for their time. Altogether, the study lasted approximately three hours.

The AGQ-R, a 12-item, self-report measure designed by Elliot and Murayama [5], was used to assess achievement goal orientation and classify students based on their AGQ-R scores. A 4-option multiple choice, 30-item test was administered before and after students interacted with MetaTutor to assess knowledge of the circulatory system. The pre- and post-tests were counterbalanced and randomized across all students. Students also completed self-report questionnaires on emotions and personality at the beginning and end of the session.

2.2 MetaTutor: An Intelligent Tutoring System that Fosters Self-Regulation While Learning About Science

MetaTutor provided students with 47 pages of content to learn about the human circulatory system [3]. Four pedagogical agents (PAs; Pam the Planner, Gavin the Guide, Mary the Monitor, Sam the Strategizer) were designed to intervene for timely scaffolding and foster effective SRL strategies by mirroring human tutor interactions. Students were presented with an overall learning goal: gain as much knowledge about the human circulatory system as possible within a 90-min timeframe. They were given the opportunity to set two sub-goals during the sub-goal setting phase with help from Pam the Planner. Students could reprioritize, add, or complete sub-goals deemed appropriate to achieve the overall learning goal at any time during the session. Each of the PAs were responsible for a specific SRL strategy and process: Mary the Monitor prompted students to monitor progress toward sub-goals (e.g., heart components); Pam the Planner helped students set sub-goals related to the overall learning goal; Sam the Strategizer intervened to prompt SRL strategy use such as taking notes; and Gavin the Guide introduced learners to the system and administered several self-report questionnaires on emotions during the session.

The MetaTutor interface displayed the overall learning goal and set sub-goals at the top of the screen (see Fig. 1). When students employed SRL strategies and processes, they indicated doing so on the SRL palette at the middle-right side of the screen. All system-initiated actions (e.g., Sam the Strategizer prompting students to summarize) were strategically designed to intervene when certain conditions were met (e.g., Sam the Strategizer prompted students to summarize after reading based on a time threshold; Mary the Monitor prompted students to assess the relevancy of a content page if they were reading text that was irrelevant to their sub-goal).



Your goal is to learn all you can about the Circulatory System. Specifically, be sure to learn about all the different organs and other components of the circulatory system, and their purpose within the system, how they work both individually and together, and how they support the healthy functioning of the body.

Your current subgoals are
Heart components
Purposes of the circulatory system
Complete Subgoal
Prioritize Subgoal
Add New Subgoal
Anatomy and Structure of the Heart
See Contents in Full View

Anatomy and Structure of the Heart

The human heart is one of the most vital organs in the human body. It acts as a pump to continuously circulate blood throughout the body. The human heart weighs approximately 250-350 grams.

The heart is enclosed in a double walled sac called the pericardium. This sac provides protection to the heart. Pericardial fluid is found in the pericardial cavity between the two walls of the pericardium; this fluid lubricates and cushions the heart as it beats. The outer wall of the pericardium serves to protect the heart and prevents the heart from overfilling with blood while it is pumping. The inner wall of the pericardium provides lubrication so that the heart can move smoothly when it contracts with each heartbeat.

The human heart organ is composed of three layers and is surrounded by the pericardium. The inner layer of the heart is called the endocardium. This layer lines each chamber of the heart and comes in direct contact with the circulating blood. The middle layer of the heart, known as the myocardium, is composed of specialized cardiac muscle which can contract. The outermost layer of the heart is known as the epicardium. It is this outer layer that interacts with the pericardium.

Sam the Strategizer
I would like to:
Plan my learning by...
Telling what I already know about this
Monitor my learning by...
Assessing how well I understand this
Evaluating how well I already know this content
Evaluating how well this content matches my current subgoal
Apply a learning strategy:
Take notes
Make an inference
Summarize
Take Survey Now
StatusLog
Pause Metatutor

Fig. 1. Screenshot of the MetaTutor interface.

At the beginning of the session, students were assigned to one of two conditions: prompt and feedback or the control condition. In the control condition, students did not receive any feedback or interventions from PAs to foster scaffolding and initiate SRL strategies and processes, while students assigned to the prompt and feedback condition were consistently prompted by PAs to employ SRL strategies and processes as described in the above section, and were given feedback on their use of these processes. Given the PA interventions and prompts in the experimental group could have contributed to a prolonged session duration relative to the control condition, our analyses controlled for time spent in the session to eliminate bias. Since the conditions were not considered in the research questions or analyses, limited details are provided to maintain concision.

2.3 Experimental Procedure

During the first day of the study, students were randomly assigned to one of the two conditions. Following their consent (i.e., consent form), students were instrumented and instructed to sit in front of a computer to calibrate their eye tracking and establish a baseline for the electrodermal activity (EDA) bracelet and facial recognition software. Students were then presented with demographic questions and several self-report questionnaires to gauge their emotions, personality, and motivational drive. Next, a 30-item, multiple choice pre-test to assess prior knowledge of the circulatory system was administered. This part of the study typically lasted 30 to 60 min.

On the second day of the study, students' eye tracking was calibrated and their baselines were established for the facial recognition software and EDA bracelet prior to beginning the session with MetaTutor. When students first interacted with the ITS, they were introduced to the system by Gavin the Guide who displayed tutorials for navigational purposes and introduced the other PAs. Afterwards, they were presented with the overall learning goal and began the sub-goal setting phase, where Pam the Planner prompted students to set two sub-goals and provided feedback on their pre-test performance relative to all sub-goals, giving students a chance to change sub-goals they already set based on where they needed more improvement. Next, students were able to select pages from the table of contents and begin working toward their first sub-goal. Throughout the 90-min session, students could indicate when they were going to employ a self-regulatory strategy or process by using the SRL palette. Once they completed the learning session, students were directed to a 30-item, multiple choice post-test to assess knowledge gained about the circulatory system after interacting with MetaTutor and completed self-report questionnaires to measure emotions and reactions toward the PAs. Students were then paid, debriefed and thanked for their time.

2.4 Data Coding and Scoring

A formula developed by Witherspoon et al. [8] was calculated to measure proportional learning gain (PLG), a representation of improvement at post-test with consideration of pre-test performance. Separate PLG scores were calculated for sub-goals 1 and 2 based on questions answered correctly on the pre- ($M = 17.25$, $SD = 4.41$, range = 7–28) and post-tests ($M = 20.61$, $SD = 4.24$, range = 5–29) that were related to the corresponding sub-goal. Separate PLG scores were calculated because some questions were not related to all sub-goals. For instance, question 1 may have been related to sub-goal one (e.g., heart components) since it dealt with the aortic valve, but may not have been relevant to sub-goal two (e.g., blood vessels); therefore, questions unrelated to a particular sub-goal were not included in the PLG calculation for said sub-goal.

Cognitive strategy use was based on the total number of user-initiated actions for summaries (SUMM), inferences (INF), note taking, (TN), and prior knowledge activation (PKA) for each student. Metacognitive process use was calculated based on the number of times students used the SRL palette to indicate they made judgments of how much they were understanding the content (JOL), feelings of knowing the material they were interacting with (FOK), and evaluating the content they were reading to gauge relevancy to the sub-goal they were working to complete (CE). In addition, students monitored their progress toward achieving sub-goals (MPTG) by clicking on the 'complete sub-goal' button beside the sub-goal progress bar. All system-initiated actions (i.e., PA intervention) were not considered in the analyses to examine authentic SRL strategy deployment. These data were extracted via log files.

The independent variables were based on the AGQ-R, which operationalizes motivation as a 2 × 2 framework and describes four goal orientation constructs: mastery, performance, approach and avoidance. Students receive a total of four scores for all combinations of each dimension (e.g., mastery-approach, performance-avoidance, etc.) where each score ranged from 1–30. Previous research revealed inconsistent results on learning when considering only the approach and avoidance dimension of goal orientation [7]; therefore, to explore how meaningful each component is, we summed each construct into four scores: mastery, performance, approach, and avoidance and created two independent variables: one for mastery ($n = 73$), performance ($n = 50$), and a combination of mastery and performance ($n = 67$) and the other variable included approach ($n = 95$), avoidance ($n = 22$), and a combination of approach and avoidance ($n = 73$) goal orientations. Students were assigned to a group in each dimension (e.g., mastery or performance); however, if students had a score on either dimension less than 2 points different from the other dimension, they were assigned to the combination group for that dimension. For example, if a student received a score of 14 on mastery and a 16 on performance, they were assigned to the performance group. If a student received a score of 14 on mastery and a 15 on performance, they were assigned to the combination group for the mastery and performance dimension, and same with the approach and avoidance dimension, to assess whether students can be orientated in more than one dimension and how appropriate the combination categorization is.

3 Results

3.1 Research Question 1: Are There Significant Differences Between Goal Orientation Groups on Sub-goal 1 PLG and Sub-goal 2 PLG, While Controlling for Session Duration?

A MANCOVA was calculated to examine the role of two independent variables for dimensions of goal orientation on sub-goal 1 PLG and sub-goal 2 PLG, while controlling for session duration. Pillai's trace (V) was used due to the unequal sample sizes between the groups. Repeated measure analyses were not used because students set different sub-goals throughout the session, given the autonomy of selecting from a list of seven sub-goals (e.g., one student may set two different sub-goals compared to another student). Our analyses included the first two sub-goals rather than all of the sub-goals set since it is a requirement of the ITS for students to set two sub-goals prior to interacting with MetaTutor and not all students may set the same number of sub-goals after the sub-goal setting phase. The results revealed a significant main effect for approach, avoidance, and approach and avoidance combination groups, while controlling for session duration, $F(4, 360) = 4.60, p = .001$, Pillai's V = .10, $\eta_p^2 = .05$, but not for mastery, performance, and mastery and performance combination groups, $F(4, 360) = .93, p = .45$, Pillai's V = .02, $\eta_p^2 = .01$. In addition, there was no significant interaction effect, $F(8, 360) = .57$, $p = .80$, Pillai's V = .03, $\eta_p^2 = .01$.

Between-subject analyses revealed significant differences between approach, avoidance and combination groups based on PLG for the first sub-goal set, $F(2, 180) = 8.82$, $p = .00$, $\eta_p^2 = .09$, but indicated no significant differences between approach, avoidance, and combination groups based on PLG for the second sub-goal, $F(2, 180) = .03$, $p = .973$, $\eta_p^2 = .01$. More specifically, students in the combination group achieved the highest average PLG based on the first sub-goal (refer to Table 1 for the means and standard deviations between the different groups) compared to the remaining groups.

Table 1. Means, adjusted means, standard deviations, and standard errors between approach, avoidance, and combination groups on sub-goal 1 PLG and sub-goal 2 PLG.

Group	Performance			
	SG 1 PLG		SG 2 PLG	
	M (*SD*)	*M*M_{adj} (*SE*)	*M* (*SD*)	M_{adj} (*SE*)
Approach	27.32 (50.79)	27.08 (5.98)	24.76 (63.27)	24.45 (6.24)
Avoidance	−29.80 (98.16)	−29.31 (12.43)	26.21(37.54)	26.846 (12.95)
Combination	29.25 (50.91)	29.42 (6.82)	20.68 (62.57)	20.90 (7.11)

Note. PLG = proportional learning gain; SG = sub-goal.

3.2 Research Question 2: Are There Significant Differences Between Goal Orientation Groups on Cognitive Strategy Use, While Controlling for Session Duration?

A MANCOVA was calculated with two independent variables: mastery, performance, combination and approach, avoidance, combination groups of goal orientation on user-initiated cognitive strategy use, while controlling for session duration. The analysis indicated no statistically significant differences between mastery, performance, and combination group on user-initiated cognitive strategy use, while controlling for session duration, $F(8, 356) = 1.31$, $p = .238$, Pillai's V = .06, $\eta_p^2 = .03$ or approach, avoidance, and combination group on user-initiated cognitive strategy use, while controlling for session duration, $F(8, 356) = .90$ $p = .518$, Pillai's V = .04, $\eta_p^2 = .02$. See Table 2 for adjusted means and standard errors.

Table 2. Means, adjusted means, standard deviations, and standard errors between goal orientation groups on cognitive strategy use.

Group	Cognitive strategy			
	SUMM	INF	TN	PKA
	M_{adj} (*SE*)	M_{adj} (*SE*)	M_{adj} (*SE*)	M_{adj} (*SE*)
Approach	1.45 (.29)	.35 (.06)	11.78 (1.47)	.21 (.06)
Avoidance	.98 (.61)	.19 (.13)	10.02 (3.05)	.27 (.13)
CombinationAA	1.57 (.33)	.31 (.07)	10.59 (1.67)	.26 (.07)
Mastery	1.26 (.33)	.30 (.07)	13.74 (1.65)	.22 (.07)
Performance	1.45 (.40)	.44 (.08)	12.33 (1.98)	.30 (.09)
CombinationMP	1.63 (.35)	.24 (.07)	7.36 (1.72)	.21 (.08)

Note. CombinationAA = combination of approach and avoidance dimensions of goal orientation; CombinationMP = combination of mastery and performance dimensions of goal orientation; SUMM = summarizing; INF = inferences; TN = note-taking; PKA = prior knowledge activation.

3.3 Research Question 3: Are There Significant Differences Between Goal Orientation Groups on Metacognitive Strategy Use, While Controlling for Session Duration?

A MANCOVA was used to assess the influence of mastery, performance, combination and approach, avoidance, combination groups of goal orientation on user-initiated metacognitive processes, while controlling for session duration. There were no significant differences between mastery, performance, and combination group, $F(8, 356) = 1.31$, $p = .238$, Pillai's V = .06, $\eta_p^2 = .03$ and approach, avoidance, and combination groups, $F(8, 356) = .90$, $p = .518$, Pillai's V = .04, $\eta_p^2 = .02$ on user-initiated metacognitive processes, while controlling for session duration. See Table 3 for adjusted means and standard errors.

Table 3. Means, adjusted means, standard deviations, and standard errors between goal orientation groups on metacognitive strategy use.

Group	Metacognitive processes			
	MPTG	CE	FOK	JOL
	M_{adj} (*SE*)	M_{adj} (*SE*)	M_{adj} (*SE*)	M_{adj} (*SE*)
Approach	3.62 (.28)	1.02 (.29)	1.87 (.35)	3.61 (.71)
Avoidance	3.96 (.58)	.57 (.61)	.69 (.72)	3.17 (1.47)
CombinationAA	3.91 (.32)	.54 (.34)	.73 (.40)	3.42 (.81)
Mastery	3.56 (.32)	.63 (.34)	1.11 (.40)	2.58 (.81)
Performance	3.88 (.38)	.47 (.40)	1.40 (.49)	4.44 (.97)
CombinationMP	3.93 (.33)	1.19 (.35)	1.42 (.42)	3.75 (.84)

Note. CombinationAA = combination of approach and avoidance dimensions of goal orientation; CombinationMP = combination of mastery and performance dimensions of goal orientation; MPTG = monitoring progress toward goals; CE = content evaluation; JOL = judgment of learning; FOK = feeling of knowing.

4 Discussion and Implications for Designing ITSs

Overall, results revealed differences in proportional learning gain (PLG) based on the first sub-goal set during the sub-goal setting phase between approach, avoidance, and approach and avoidance combination groups, but not for mastery, performance and mastery and performance combination groups. Specifically, we found students who adopted a combination of both approach and avoidance components of goal orientation had the highest PLG based on questions answered correctly on the pre- and post-tests related to sub-goal 1. In contrast, there were no differences in PLG related to sub-goal 2 among the goal orientation groups. The results do not support our hypotheses where we suspected differences between goal orientation groups on sub-goal PLG. One possible explanation could be due to changes in goal orientation following sub-goal 1, where students who adopted a particular goal orientation at sub-goal 1 and changed to another goal orientation as they continued to work towards sub-goal 2. Future studies should administer the AGQ-R periodically throughout a learning session (e.g., before students start a new sub-goal) to assess potential changes in goal orientation.

Follow up analyses support changes in goal orientation as well: there were no differences among goal orientation groups on cognitive strategy use and metacognitive processes, which is not consistent with literature [e.g., 4, 6]. Any differences in cognitive and metacognitive processes and strategy use between the groups may not have been recorded if students had switched to another goal orientation, since the AGQ-R is only administered *once* and recognizes them as belonging to their original classification throughout the learning session, resulting in a lack of differences between groups. From a methodological perspective, researchers and ITS developers should focus on the quantitative (e.g., frequency of use across time) and quality (e.g., shift from low-level cognitive strategies such as taking notes to high-level strategies like making inferences) of cognitive and metacognitive strategy use during learning with ITSs as evidence of changes in motivational processes.

The results lead us to believe additional analyses of trace data (e.g., comparing specific sub-goal durations over the learning session, extracting SRL strategy use during specific sub-goals and measuring the quality of cognitive and metacognitive SRL strategies and processes used between goal orientation groups) and *re-defining motivation as a dynamic variable that fluctuates over time and across contexts* [9] should be considered in future studies and the design of ITSs. In addition, our findings could have implications for the design of adaptive ITSs that focus on the individualized CAMM needs of the student. Using multiple data channels over durations of various tasks to measure motivational processes (e.g., task value, self-efficacy) could reveal patterns in data for why and when students need virtual agent intervention to foster effective SRL strategy use and promote learning based on their individual, motivational needs. For example, trace data from log files on cognitive strategy use can reveal where students are failing to effectively use strategies, and therefore, PAs could focus on developing students' self-efficacy by modeling strategy use, providing supportive feedback, and encouraging strategy use during a task. Researchers should also consider du Boulay and del Soldato's [10] who suggest focusing on the role of values in motivation to reveal specific reasons for why students are unmotivated or motivated during a task. The implications of developing ITSs to measure motivation based on multiple data channels could enhance learning outcomes and extend models of computerized tutoring. Thus, exploring motivation and which data channels are most indicative of motivational states could lead ITS design in the direction of adaptive interventions and feedback based on what personally motivates students to learn.

Acknowledgements. This research was supported by funding from the National Science Foundation (DRL#1431552; DRL#1660878, DRL#1661202) and the Social Sciences and Humanities Research Council of Canada (SSHRC 895-2011-1006). Any opinions, findings, conclusions, or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation or Social Sciences and Humanities Research Council of Canada. The authors would like to thank the members from the SMART Lab at NCSU for their assistance with data collection.

References

1. Azevedo, R., Taub, M., Mudrick, N.V.: Understanding and reasoning about real-time cognitive, affective, and metacognitive processes to foster self-regulation with advanced learning technologies. In: Schunk, D.H., Greene, J.A. (eds.) Handbook on Self-Regulation of Learning and Performance, 2nd edn, pp. 254–270. Routledge, New York (2018)
2. Winne, P.H., Hadwin, A.F.: The weave of motivation and self-regulated learning. In: Schunk, D., Zimmerman, B. (eds.) Motivation and Self-Regulated Learning: Theory, Research, and Applications, pp. 297–314. Erlbaum, Mahwah (2008)
3. Azevedo, R., Harley, J., Trevors, G., Duffy, M., Feyzi-Behnagh, R., Bouchet, F., Landis, R.: Using trace data to examine the complex roles of cognitive, metacognitive, and emotional self-regulatory processes during learning with multi-agent systems. In: Azevedo, R., Aleven, V. (eds.) International Handbook of Metacognition and Learning Technologies. SIHE, vol. 28, pp. 427–449. Springer, New York (2013). https://doi.org/10.1007/978-1-4419-5546-3_28
4. Zimmerman, B.J., Schunk, D.H.: An essential dimension of self-regulated learning. In: Schunk, D.H., Zimmerman, B.J. (eds.) Motivation and Self-Regulated Learning: Theory, Research, and Applications, pp. 1–30. Taylor & Francis Group, LLC, New York (2008)
5. Elliot, A.J., Murayama, K.: On the measurement of achievement goals: critique, illustration, and application. J. Educ. Psychol. **100**(3), 613–628 (2008)
6. Coutinho, S.A., Neuman, G.: A model of metacognition, achievement goal orientation, learning style and self-efficacy. Learn. Environ. Res. **11**(2), 131–151 (2008)
7. Vaessen, B., Prins, F., Jeuring, J.: University students' achievement goals and help seeking strategies in an intelligent tutoring system. Comput. Educ. **72**(31), 196–208 (2014)
8. Witherspoon, A.M., Azevedo, R., D'Mello, S.: The dynamics of self-regulatory processes within self-and externally regulated learning episodes during complex science learning with hypermedia. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) ITS 2008. LNCS, vol. 5091, pp. 260–269. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-69132-7_30
9. Winne, P.H.: Cognition and metacognition in self-regulated learning. In: Schunk, D., Greene, J. (eds.) Handbook of Self-Regulation of Learning and Performance, pp. 36–48. New Routledge, New York (2017)
10. Du Boulay, B., del Soldato, T.: Implementation of motivational tactics in tutoring systems: 20 years on. Int. J. Artif. Intell. Educ. **26**(1), 170–182 (2016)

Game Scenes Evaluation and Player's Dominant Emotion Prediction

René Doumbouya, Mohamed S. Benlamine(✉), Aude Dufresne, and Claude Frasson

Heron Lab, Department of Computer Science, University of Montreal, Montreal, Canada
{rene.lacine.doumbouya,ms.benlamine,dufresne}@umontreal.ca,
frasson@iro.umontreal.ca

Abstract. In this paper, we present a solution for computer assisted emotional analysis of game session. The proposed approach combines eye movements and facial expressions to annotate the perceived game objects with the expressed dominate emotions. Moreover, our system EMOGRAPH (Emotional Graph) gives easy access to information about user experience and predicts player's emotions. The prediction mainly uses both subjective measures through questionnaire and objective measures through brain wave activity (electroencephalography - EEG) combined with eye tracking data. EMOGRAPH's method was experimented on 21 participants playing horror game "*Outlast*". Our results show the effectiveness of our method in the identification of the emotions and their triggers. We also present our emotion prediction approach using game scene's design goal (defined by OCC variables from the model of emotions' cognitive evaluation of Ortony, Clore and Collins [1]) to annotate the player's situation in a scene and machine learning algorithms. The prediction results are promising and would widen possibilities in game design.

Keywords: Affective computing · Video games · Player model · EEG
Cognitive evaluation (OCC)

1 Introduction

Nowadays, thanks to technological advances, video games continue to grow and become based on more sophisticated rules and more realistic elements to captivate and engage players during interactions. To advance this field, video game research should focus more on user experience by becoming more aware of the player's emotions. From a human-centred computing perspective, the end-user's emotional reaction within learning/gaming environments is a critical challenge to design for, with empirical evidence already showing that not all the used practices are necessarily effective from a learning/gaming perspective [2]. It is important to analyze the emotional dynamics in video games because it enhances the player's feeling of presence and immerses him in an intense emotions experience more than other media [3–5].

With scientific and technological progress, detecting emotional reaction is now more accessible via tools based on ocular devices (webcam, infrared camera, Kinect, eye tracker …) or physiological sensors (EDA, HR, EMG, EEG, Respiration rate …). With

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 54–65, 2018.
https://doi.org/10.1007/978-3-319-91464-0_6

these tools, it is possible to capture physiological data and recognize player's emotions. In the literature, emotions generation can be described by three components: cognition, physiological processes and interactions between them [6]. Nevertheless, the most applied theories are of cognitive evaluation as reported by Scherer [7] and Lazarus [8]. These theories focus on the cognitive perception of situations or events generating emotional responses. Several researchers suggest that emotional reactions are regulated by two basic motivational systems: Avoidance system and Approach system [9]. Moreover, many studies [10, 11] associate the Frontal alpha asymmetry (FAA) EEG measure with Approach and Avoidance tendencies and individual differences in personality and also other studies [12, 13] underscore the role of prefrontal cortex in emotion process.

In this paper, we aim to answer the following research questions: (1) what is the connection between the game elements (or scenes) perceived by the participant and his emotional state? And (2) can we predict the generated dominant emotion from a scene based on the characteristics of the player, his Approach/Avoidance behavior and the design objective of the scene (the emotions targeted by the designer).

To answer these questions, we propose an evaluation system using eye tracking data and facial expressions to make the association between game elements and the dominant emotion. This paper describes our resulting system that matches between the scenes during a game session, and the players' emotion. We designed experiments with 21 participants who played a commercial horror game: *Outlast*. During the game session, the participant was equipped with webcam based facial expressions detection tool, recording his emotions and an eye-tracker recording his gaze on the screen. To address the prediction issue, we used a theoretical model of cognitive evaluation of emotions Ortony, Clore and Collins (OCC) [1] for the annotation of the player' situation in the scene enriched with the player characteristic (socio-demographic information, and personality trait) as an input to the Machine Learning algorithm to predict the player's dominant emotion when interacting with a game scene.

2 Background on Video Games and Emotion Analysis

The reactions of a person playing are different from one game to another and the design of video games is able to influence the affective state of the player [14–17].

2.1 Game Elements and Generated Emotions

It's more advantageous to make video game design more centered on players' emotions. In fact, the players' actions and feelings are not the same even if we put them in the same game situation. Player's emotions need to be analyzed when interacting with game elements (aesthetics, visual and audio elements) during a game session. Therefore, it is interesting in our research to understand: "*what are the precise elements in the interface that the gamer is watching and interacting with*". For that reason, we have opted to use eye-tracking techniques in this study.

2.2 Measuring Emotions

Used techniques can be categorized on subjective and objective measures.

- Subjective measures consist of self-evaluation of the person's emotions; the users report their own emotions. Different types of subjective measures can be used open-ended, multiple-choice, and Likert-scales items in surveys. As a famous example, we can cite the Self-Assessment Manikin [18].
- Objective measures consist of capturing and analyzing the signals coming from the player's body and face. Different tools can be used such as cerebral activity EEG [19], skin conductance [20] or facial expressions recognition [21].

2.3 Learning with Intense Emotions Evoking Games

Learning is a process that always involves emotional component because it is mostly treated in the limbic system (especially Hippocampus, Thalamus, Hypothalamus and Amygdala) [22, 23]. In pedagogy, teachers are encouraged to instill an environment that promotes positive emotions to activate the hippocampus (for information processing and transfer to the prefrontal cortex) rather than to create an environment of fear and stress - thus activating the amygdala (Defensive survival circuit) [22, 24]. In games (e.g. MMORPG, adventure and horror games), in situations like facing hard enemies, the use of violent messages to push the player to the limit may lead to the success of the team. Teacher sometimes requires laying down strict guidelines which is pedagogically not advised if used directly in the classroom. So game agents playing the role of teacher can use some directive language (maybe sarcastic) to push students reach challenging levels in a game where they have fictive pseudonyms (to make things not personal). This method may lead the students to adopt the group's objective and do more practice to meet that objective (guided by the teacher) because in such environment players accept to be exposed in such situations and may enjoy it [25, 26]. This field is less explored in educational research and needs more attention.

3 Experimental Settings

3.1 Participants

This study involves 21 participants (12 males; 9 female), aged between 18 and 35 years from a North American university. We have discarded 2 participants due to technical problem while collecting data. We categorized the players according to the number of hours of play per week (5 extreme players, 6 intermediates and 8 novices).

3.2 The Game - Outlast

In this study, participants were asked to play the first level of a horror commercial game named *Outlast* (developed by Red Barrels Games). Outlast allows players to embody Miles Upshur, an investigative reporter sent to find the truth about the company Murkoff

Corporation. As a reporter, the player must find a way to enter the asylum before starting his investigation. The only means of the player's survival are: "flee or hide" armed only with his camera. The player is placed in strange game situations that influence his emotional reactions. The participants were required to play the first stage of the game.

3.3 Experiment and Equipment

The experimental scheme is presented in Fig. 1 showing the positions of the sensors placed on the body (right) and those integrated in the experimental computer (left). In this study, we use facial expression recognition module combined with electroencephalogram (EEG) and the eye-tracking systems.

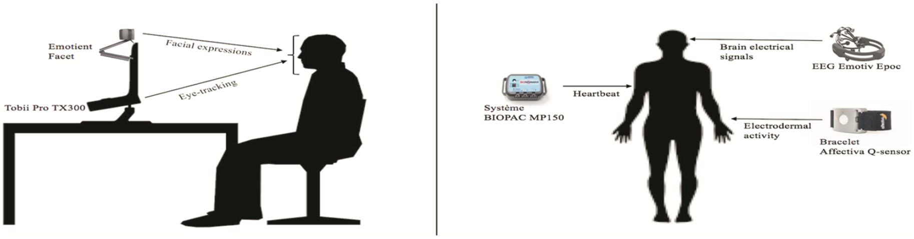


Fig. 1. Experimental design

3.4 Measures

Different studies have used several methods, subjective or objective, in order to evaluate the emotion in game scenes.

Subjective measures. To gather as much information as possible to complete our study, forms were submitted to the participants before and after the game. Two forms were filled before session: the first questionnaire for collecting socio-demographic data (age, gender and ethnicity), school level and hours of play per week and the second was the "Big Five" questionnaire [27] for the assessment of the participant's personality traits. In the post-test, we used the immersion and flow questionnaire. The questionnaire was adapted from the GameFlow questionnaire [16]. In addition, participants were asked to answer a final questionnaire about their emotions felt during stages of the game.

Objective measures. We collected multimodal data from the player's body and face to analyze his affective and mental state. In this study, we are using among these measures: EEG data to compute the *Frontal Alpha Asymmetry* (FAA), facial expressions to extract dominant emotion and eye-tracking data to detect the scenes visualization time and duration.

Frontal Alpha Asymmetry. Neuroscientists have found that higher alpha power (8–12 Hz) of the left compared to the right frontal brain is related to positive feelings [28].

Facial expression analysis. The iMotions FACET software generates numerical values for each basic emotion. These values scaled between [– 5, 5] are the log likelihood of the presence of an emotion. For example, a joy value of 4 means a big smile and any human expert would see that. A joy value of 0 means that the observed expression is equally categorized by human experts as "*being*" and "*not being*" joyful.

Eye-tracking analysis. We used the software iMotions to replay and annotate chronologically the participants' game session by defining Areas of Interest (AOI) in order to get gaze statistics by AOI. Participants may take different durations for each scene during their course in the game. Using hit time and the time spent metrics, we identified the time and duration that the player spent for each scene.

4 Method

Multimodal analysis was performed using several sources of information to determine players' emotional reactions. This require the use of statistical analyzes and AI techniques to construct the player's dominant emotion model.

4.1 Dominant Emotion Extraction

Even if the player has a fixed path in the game, the participants don't look at all the AOIs. The participant's dominant emotion is identified using the following steps:

- Every time window of 500 ms [29], calculate the median of each emotion and assign 1 to the emotion if its median exceeds a relative threshold otherwise 0.
- For each AOI, sum the binary scores reported for each emotion.
- Select the emotion with the highest score: Multiple emotions can occur in the same time window, we attribute the score "1" to the emotion whose median is the highest. In the case of having equal emotional counts for an AOI, we choose as dominant emotion from the player emotion self-report.

At the end, EMOGRAPH stores, in a MySQL database, information about the participant, visualized AOIs, and the corresponding dominant emotion. This database is used by the player's experience visualization and the emotion prediction modules.

4.2 The FAA Computation

The frontal asymmetry index was computed from raw frontal EEG data using electrodes F3/F7 and F4/F8. We calculated FAA using the formula below:

$$FAA = \log\left(\frac{Alpha\ Power_{Right} - Alpha\ Power_{Left}}{Alpha\ Power_{Right} + Alpha\ Power_{Left}}\right)$$

Higher scores on this asymmetry index indicate greater relative left hemisphere activation which means that the player's behavior in the scene is APPROACH otherwise it is AVOIDANCE.

4.3 OCC Game Scene Representation

To predict the player dominant emotion for new game scenes, we need first to characterize the game scenes according to the designer perception and goals using variables from OCC model [1].

Cognitive variables characterize a person's interpretation of a situation. In fact, the emotional response depends on the situation interpretation as desirable or undesirable, expected or unexpected, etc. The OCC model has its own descriptive variables divided into 3 categories: global, central and local variables. Global variables are included in all situations, whereas the central and local variables are specific to a certain situation characterizing their informational content. This model identifies the involved variables and allows the description of a game scene by precise variables summarized in table above (see Table 1). The scene OCC representation is applicable to all kind of games intended to learning or entertainment.

Table 1. Global, central and local variables and their associated values

	Global variables		Central variables			Local variables										
Evaluation Variable	Surprise	Sense of reality	Desirability	Approval	Attraction	Desirability by other	Esteem for other	Merit for other	Likelihood	Realization	Effort	Agent	Power of the link	Deviation	Disposition	Familiarity
Values	0 (False) or 1 (True)	0 (False) or 1 (True)	-1, -0.5, 0, 0.5, 1	-1, -0.5, 0, 0.5, 1	-1, -0.5, 0, 0.5, 1	-1, -0.5, 0, 0.5, 1	-1, -0.5, 0, 0.5, 1	-1, -0.5, 0, 0.5, 1	-1, -0.5, 0, 0.5, 1	-1, -0.5, 0, 0.5, 1	-1, -0.5, 0, 0.5, 1	0 (other) or 1 (self)	0, 0.5, 1	-1, -0.5, 0, 0.5, 1	0, 0.5, 1	-1, -0.5, 0, 0.5, 1

4.4 The Dominant Emotion Prediction

In this section we describe our approach to training and evaluating classifiers for the task of detecting the player dominant emotion given formal description of a scene, Approach/Avoidance player behavior and his demographical data.

Training set construction. In order to train a scene's dominant emotion predictive model, we used a training set containing scene descriptions, participant's socio-demographic information, personality traits, Approach/Avoidance behavior in scene and also the participant's dominant emotion. In our study, we targeted two objectives for predicting the dominant emotion of:

1. An existing player having as input description variables of a new scene defined by the designer.
2. A new player (defined by his socio-demographic information, player category, personality trait and Approach/Avoidance behavior) in front of an existing scene.

For each of the approaches, the models use 27-dimensions vector: 16 variables from the OCC model, 4 socio-demographic variables (gender, age, ethnicity, and player category), 5 personality trait variables, 1 variable for Approach/Avoidance and the dominant emotion. The approaches have been developed with Python language and scikit-learn library.

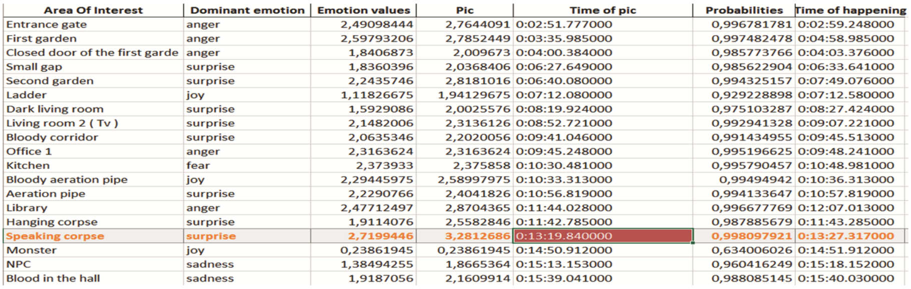
Individual approach. The prediction is based on the emotional reactions of individuals during interaction with game scene. The trained model is individual, meaning that it is specific to the participant or the scene depending on the objective of the model. We have trained and validated our models using the leave-one-out cross-validation (LOOCV) method because of the size of the individual dataset.

General approach. The prediction is based on a unique dataset (contains 335 examples) gathering all the emotional responses of participants in all scenes. We have trained and validated our model using the k-fold cross-validation method. We found the optimal number of blocks with a grid search.

5 Results

5.1 Game Analysis Results

Here above, the Fig. 2 shows the dominant emotions of participant (P21) obtained from the EMOGRAPH system. On this example it is a woman, beginner in video games. We see also the moment of appearance of the peak. For example, the orange colored line shows that the dominant emotion determined for the stage "Speaking corpse" is the surprise and the relative time of occurrence of the peak is at 00:13:19:8400 from the beginning of the game session' video. A screenshot of participant P21 at the "speaking corpse" step, illustrated by Fig. 3, is representing the orange line in Fig. 2. On this capture, we note the expression of surprise on the face of the participant when seeing the hanged corpse marked by his gaze. At the bottom, on the right, the time of the video zoomed and circled in red is equivalent to that recorded by EMOGRAPH (Fig. 3). EMOGRAPH offers the visualization of the dominant emotion frame extracted from the game session video using the peak time. These game screen-captures help game designer to better analyze the player experience.



Area Of Interest	Dominant emotion	Emotion values	Pic	Time of pic	Probabilities	Time of happening
Entrance gate	anger	2,49098444	2,7644091	0:02:51.777000	0,996781781	0:02:59.248000
First garden	anger	2,59793206	2,7852449	0:03:35.985000	0,997482478	0:04:58.985000
Closed door of the first garde	anger	1,8406873	2,009673	0:04:00.384000	0,985773766	0:04:03.376000
Small gap	surprise	1,8360396	2,0368406	0:06:27.649000	0,985622904	0:06:33.641000
Second garden	surprise	2,2435746	2,8181016	0:06:40.080000	0,994325157	0:07:49.076000
Ladder	joy	1,11826675	1,94129675	0:07:12.080000	0,929228898	0:07:12.580000
Dark living room	surprise	1,5929086	2,0025576	0:08:19.924000	0,975103287	0:08:27.424000
Living room 2 (Tv)	surprise	2,1482006	2,3136126	0:08:52.721000	0,992941328	0:09:07.221000
Bloody corridor	surprise	2,0635346	2,2020056	0:09:41.046000	0,991434955	0:09:45.513000
Office 1	anger	2,3163624	2,3163624	0:09:45.248000	0,995196625	0:09:48.241000
Kitchen	fear	2,373933	2,375858	0:10:30.481000	0,995790457	0:10:48.981000
Bloody aeration pipe	joy	2,29445975	2,58997975	0:10:33.313000	0,99494942	0:10:36.313000
Aeration pipe	surprise	2,2290766	2,4041826	0:10:56.819000	0,994133647	0:10:57.819000
Library	anger	2,47712497	2,8704365	0:11:44.028000	0,996677769	0:12:07.013000
Hanging corpse	surprise	1,9114076	2,5582846	0:11:42.785000	0,987885679	0:11:43.285000
Speaking corpse	surprise	2,7199446	3,2812686	0:13:19.840000	0,998097921	0:13:27.317000
Monster	joy	0,23861945	0,23861945	0:14:50.912000	0,634006026	0:14:51.912000
NPC	sadness	1,38494255	1,8665364	0:15:13.153000	0,960416249	0:15:18.152000
Blood in the hall	sadness	1,9187056	2,1609914	0:15:39.041000	0,988085145	0:15:40.030000

Fig. 2. Dominant emotions for participant P21

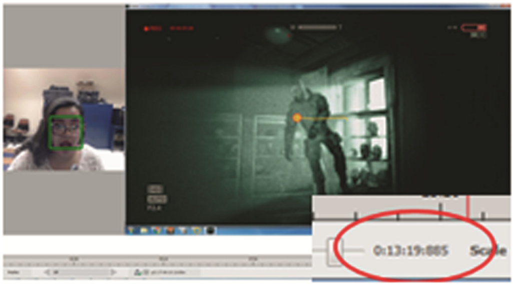
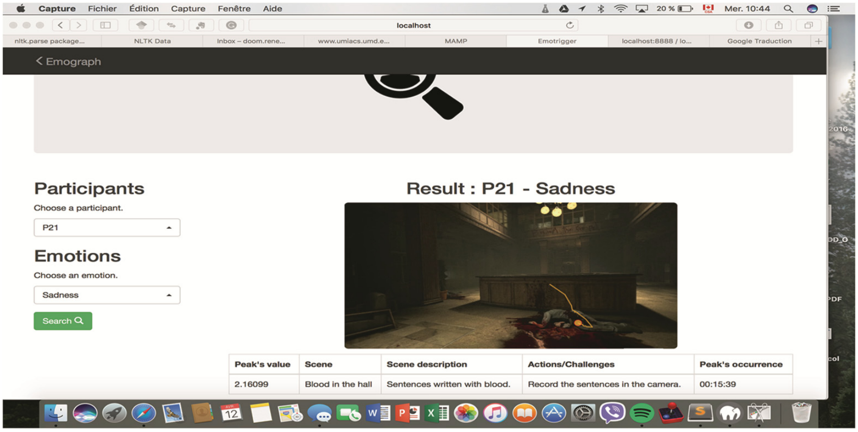


Fig. 3. Game screenshot of participant P21

EMOGRAPH: System interface.

We present the interfaces and the operating mode of the application's modules. Figure 4 shows the emotional analysis module offering to the game designer the possibility of knowing the player's dominant emotions and gives additional information about the scene. The interface proposes a search by participants or by AOI. The visualization



Participants

Choose a participant.

P21

Emotions

Choose an emotion.

Sadness

Search

Result : P21 - Sadness

Peak's value	Scene	Scene description	Actions/Challenges	Peak's occurrence
2.16099	Blood in the hall	Sentences written with blood.	Record the sentences in the camera.	00:15:39

Fig. 4. Emotional analysis module

module that displays the emotional graph is presented in Fig. 5. Additionally, by clicking on a node in the visualized emotional graph, it shows a fragment of 30 s extracted from the game-session video at the time of the dominant emotion.

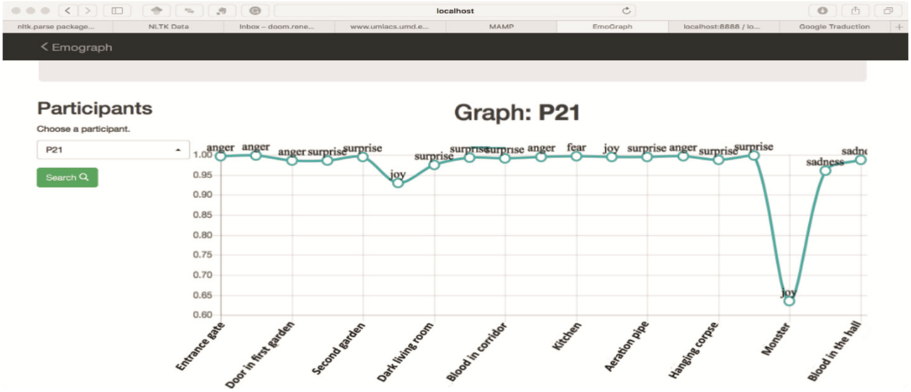


Fig. 5. Emotional transition graph module

The emotional transition graph emphasizes the dominant emotions related to game scenes. The emotions values are presented graphically after being transformed into probabilities (between 0 and 1), which facilitates their interpretation and presentation. This transformation is made according to the following formula:$PP = \dfrac{1}{1 + 10e^{-LLR}}$ with LLR being the numerical value (evidence) of the considered emotion. Figure 5 shows the emotion graph for participant P21 displayed by our system. The scenes in the game sequence are on the abscissa axis while the probabilities are on the ordinate axis. Each point gives the dominant emotion of the scene.

5.2 Prediction Results

Individual approach. For the distance-weighted k-NN algorithm, the validation method LOOCV allows each participant to choose the optimal number of neighbors (k) by testing the performance over several values of k varying from 2 to 7. For the random forests algorithm, we found the optimal number of examples by leaf using the same method.

Objective 1: Existing Person vs New Scene. The performance of our model is evaluated individually; the accuracy varies from one participant to another. Accuracy scores range from 42% to 90%. The average accuracy on all participants is 85% with the distance-weighted k-NN algorithm. Accuracy ranges from a minimum of 75% to a maximum of 95%. The average accuracy is 90%, in terms of the random forest algorithm.

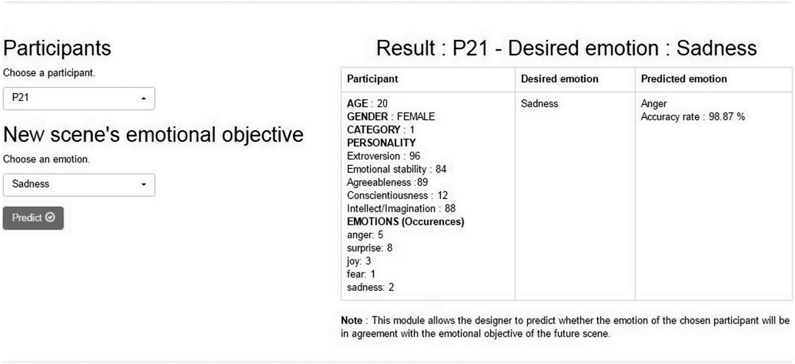
Objective 2: Existing Scene vs New Person. For this purpose, precision varies from scene to scene with scores ranging from 27% to 59% accuracy. The average accuracy on all scenes is 30% with the distance-weighted k-NN algorithm. The accuracy ranges from 40% to a maximum of 80%. The average accuracy is 64%, with the random forest algorithm.

General approach. In this approach, we have varied the number of blocks for validation between 2 and 10 for the distance-weighted k-NN and between 5 and 10 for random forests classifier to optimize the parameters (or Hyper-parameters) of the algorithms. For the distance-weighted k-NN algorithm, the number of blocks that gave the highest accuracy rate is K * = 2 with the optimal neighbor number (k) of k * = 4. The accuracy rate is 84%, also we have varied k between 2 and 7. For the random forest algorithm, we found an optimal number of blocks of K * = 5, an optimal number of trees equivalent to 40, the maximum number of dimensions equivalent to the square root of the total number of blocks. Dimensions, maximum tree depth, the nodes are extended until all the leaves are pure or until all the leaves contain less than 2 samples. We get an accuracy that reaches **96%.**

Table 2 summarizes the results obtained by our algorithms in the approaches. From these results, the random forest classifier algorithm has been integrated into our general approach prediction module. The EMOGRAPH's prediction module interface (Fig. 6) proposes to the designers the prediction of the player's emotion, with the reliability indicator which is the f-score recorded during training/validation.

Table 2. Summary of results from Machine Learning

	Individual approach		General approach
Validation method	LOOCV		K fold
	Objective 1	*Objective 2*	*General*
Distance weighted k-nearest neighbords	85%	30%	83% with K * = 2 and *k * = 4*
Random forest classifier	90%	64%	**96% with K * = 5**



Participants

Choose a participant.

P21

New scene's emotional objective

Choose an emotion.

Sadness

Predict

Result : P21 - Desired emotion : Sadness

Participant	Desired emotion	Predicted emotion
AGE : 20 **GENDER** : FEMALE **CATEGORY** : 1 **PERSONALITY** Extroversion : 96 Emotional stability : 84 Agreeableness :89 Conscientiousness : 12 Intellect/Imagination : 88 **EMOTIONS (Occurences)** anger: 5 surprise: 8 joy: 3 fear: 1 sadness: 2	Sadness	Anger Accuracy rate : 98.87 %

Note : This module allows the designer to predict whether the emotion of the chosen participant will be in agreement with the emotional objective of the future scene.

Fig. 6. Emotional prediction module

The EEG data are not necessary in order to use the system. Without EEG data the system will present two outputs one for the Approach case and the second for the Avoidance case. The model can be used for both educational or entertainment games, we only need scenes descriptions with the OCC variables of the planned game in the conceptualization phase, rather than a complete game. We intend that both, designers and teachers, can gain value from this system as a step towards affective recommender system that respond to design and learning objectives.

6 Conclusion

In this study, we examined the interactions between video games and player's emotions using game scene's design goals, player characteristics, EEG and eye-tracking data. We presented our method in categorizing the player's behavior as "Approach" or "Avoidance" using the Frontal Alpha Asymmetry (FAA) during time window calculated from eye tracking data. We have also built a machine learning model for predicting player's dominant emotion using game scene's design goal (defined by OCC variables), Approach/Avoidance behavior and the player's personality traits (using the Big Five questionnaire). Based on this experience, we proposed a system for emotional analysis of game session. The proposed tool allows the identification of the dominant emotion expressed by a gamer with the precise time in the scene. The integration of eye-tracking in the analysis process provides another level of accuracy for the game design. The application developed includes many features for game designers. EMOGRAPH not only produces affective analysis for game session and visualizes the player's emotional transitions, but also, the application performs emotions prediction of new person toward an analyzed game scene and for registered player toward a new game scene as well.

Acknowledgments. We thank the Natural Sciences and Engineering Research Council of Canada NSERC and BMU Games for funding this research.

References

1. Ortony, A., Clore, G.L., Collins, A.: The Cognitive Structure of Emotions. Cambridge University Press, Cambridge (1990)
2. Harley, J.M.: Measuring emotions: a survey of cutting edge methodologies used in computer-based learning environment research. Elsevier (2016)
3. Hemenover, S.H., Bowman, N.D.: Video games, emotion, and emotion regulation: expanding the scope. Ann. Int. Commun. Assoc. **42**, 1–19 (2018)
4. Villani, D., Carissoli, C., Triberti, S., Marchetti, A., Gilli, G., Riva, G.: Videogames for emotion regulation: a systematic review. Games Health J. **7**(2), 85–99 (2018)
5. Abdessalem, H.B., Frasson, C.: Real-time brain assessment for adaptive virtual reality game: a neurofeedback approach. Brain Function Assessment in Learning. LNCS (LNAI), vol. 10512, pp. 133–143. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-67615-9_12
6. Gratch, J., Marsella, S., Petta, P.: Modeling the cognitive antecedents and consequences of emotion. Cognit. Syst. Res. **10**(1), 1–5 (2009)

7. Scherer, K.R.: Vocal affect expression: a review and a model for future research. Psychol. Bull. **99**(2), 143 (1986)
8. Lazarus, R.S.: Emotion and Adaptation. Cité en1991, 9 (1991)
9. Elliot, A.J., Covington, M.V.: Approach and avoidance motivation. Educ. Psychol. Rev. **13**(2), 73–92 (2001)
10. Davidson, R.J.: What does the prefrontal cortex "do" in affect: perspectives on frontal EEG asymmetry research. Biol. Psychol. **67**(1), 219–234 (2004)
11. Amodio, D.M., Master, S.L., Yee, C.M., Taylor, S.E.: Neurocognitive components of the behavioral inhibition and activation systems: implications for theories of self-regulation. Psychophysiology **45**(1), 11–19 (2008)
12. Derbali, L., Ghali, R., Frasson, C.: Assessing motivational strategies in serious games using hidden markov models (2013)
13. Derbali, L., Frasson, C.: Prediction of players motivational states using electrophysiological measures during serious game play. IEEE (2010)
14. Kors, M.J., Ferri, G., van der Spek, E.D., Ketel, C., Schouten, B.A.: A breathtaking journey. On the design of an empathy-arousing mixed-reality game. ACM (2016)
15. Quick, J.M., Atkinson, R.K., Lin, L.: The gameplay enjoyment model. Int. J. Gaming Comput.-Mediat. Simul. (IJGCMS) **4**(4), 64–80 (2012)
16. Sweetser, P., Johnson, D.M., Wyeth, P.: Revisiting the GameFlow model with detailed heuristics. J.: Creat. Technol. **3** (2012)
17. Nacke, L.E., Kalyn, M., Lough, C., Mandryk, R.L.: Biofeedback game design: using direct and indirect physiological control to enhance game interaction. ACM (2011)
18. Bradley, M.M., Lang, P.J.: Measuring emotion: the self-assessment manikin and the semantic differential. J. Behav. Ther. Exp. Psychiatry **25**(1), 49–59 (1994)
19. Benlamine, M., Chaouachi, M., Frasson, C., Dufresne, A.: Physiology-based recognition of facial micro-expressions using EEG and identification of the relevant sensors by emotion. In: Proceedings of the 3rd International Conference on Physiological Computing Systems, PhyCS2016, vol. 1 (2016)
20. Stemmler, G.: Physiological processes during emotion. In: The Regulation of Emotion, pp. 33–70 (2004)
21. Littlewort, G., Whitehill, J., Wu, T., Fasel, I., Frank, M., Movellan, J., Bartlett, M.: The computer expression recognition toolbox (CERT). IEEE (2011)
22. Zirbel, E.L.: Learning, concept formation and conceptual change. Tufts University (2014)
23. LeDoux, J.E.: Brain mechanisms of emotion and emotional learning. Curr. Opin. Neurobiol. **2**(2), 191–197 (1992)
24. LeDoux, J.E.: Semantics, surplus meaning, and the science of fear. Trends Cognit. Sci. **21**(5), 303–306 (2017)
25. Lynch, T., Martins, N.: Nothing to fear? An analysis of college students' fear experiences with video games. J. Broadcast. Electron. Media **59**(2), 298–317 (2015)
26. Lin, J.-H. T., Wu, D.-Y., Tao, C.-C.: So scary, yet so fun: the role of self-efficacy in enjoyment of a virtual reality horror game. New Media Soc. 1461444817744850 (2017)
27. Goldberg, L.R.: The development of markers for the big-five factor structure. Psychol. Assess. **4**(1), 26 (1992)
28. Coan, J.A., Allen, J.J.: Frontal EEG asymmetry and the behavioral activation and inhibition systems. Psychophysiology **40**(1), 106–114 (2003)
29. Libet, B.: Reflections on the interaction of the mind and brain. Prog. Neurobiol. **78**(3), 322–326 (2006)

Disrupting the Rote Learning Loop: CS Majors Iterating Over Learning Modules with an Adaptive Educational Hypermedia

Muhammad Mustafa Hassan[1,2](✉) and Adnan N. Qureshi[1]

[1] University of Central Punjab, Lahore 54000, Pakistan
{Mustafa.hassan, dr.qureshi}@ucp.edu.pk
[2] University of Eastern Finland, 80101 Joensuu, Finland

Abstract. The rote learning problem has plagued the education systems of developing world since long. To name a few, improperly designed assessments, teachers' authority, rewarding verbatim answers, sheer class sizes, and individual learner differences are amongst the most notable mediators. The authors report on the design and development of an adaptive educational hypermedia, which disrupts the rote learning loop by hitting a few of the aforementioned reasons. The reported system provides a personalized learning experience to each learner, adapting on the basis of cognitive and learning styles. Further, the assessments are designed in a way that they loop each failed learning via variated paths, hence eliminating chances of rote learning. Moreover, the failed perturbations are traced back to the problematic domain segment for further knowledge acquisition. In-situ evaluations of the system with end-users (real students of Bachelor of Science in Computer Science) reveal a difference between control and experimental groups. The effect size is however moderate.

Keywords: Rote learning · Adaptive educational hypermedia AEH · Contextualization · Adaptor-innovator model · Cognitive styles Learning styles · ESL

1 Introduction

The problem of rote learning has since long plagued educational communities around the globe. The learning approaches adopted by students, in general, have direct relationship with understanding of the concepts and subsequent performance [1], and the rote learning style, in particular, degrades the quality of learned knowledge [2]. Nonetheless, rote learning—bare memorization—in itself is not a problem. It is an efficient learning technique needed in some learning contexts which need memorization [3], for example remembering alphabet of a language. However, it becomes a problem when a student adopts it as her primary learning strategy, applied to most—if not all—of her learning contexts [3]. The said student is mostly compelled to choose rote learning because it reduces cognitive load otherwise endured in understanding complex concepts, a situation commonly aroused in science subjects. However, in the long term, the knowledge learned through rote learning has less retention span when compared to other learning methods, like that of meaningful learning [4].

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 66–77, 2018.
https://doi.org/10.1007/978-3-319-91464-0_7

The recent research in educational psychology suggests that more abstract concepts are more negatively affected by rote learning approach [5]. Similarly, disciplines relating to problem solving are heavily affected by this learning technique [3]. The problem—though persistent throughout science spectrum—exhibits itself especially in mathematics and computer science, being the fields of higher abstraction applied to problem solving [6]. Further, this very mechanism of learning can even contribute to development of non-interest in science subjects [7].

Contrary to the popular belief, rote learning is not abundant in only developing countries [5]. Even after 30 years of education reforms in developed world, rote learning problem still prevails in many countries, for example in USA and Sweden [3]. The recent research tells that the students still choose rote learning when working with complex concepts [6]. However, the dynamics are different from the developing world where the pupils are trained to become rote learners from the very early age [8]. The researchers note that the educational system in developing countries, from teaching to assessment supports rote learning [9]. The students are rewarded for the reproduction and imitation of the given concepts verbatim [10]. Especially in Pakistan—where this research is carried out—it is frequently reported that learners are trained to reproduce what has been articulated to them [10]. They seldom use their own intuition in problem-solving of any kind, merely applying imitations of solutions earlier learned [8].

In this work, the authors report on the design, development, and testing of an Adaptive Educational Hypermedia (AEH) system to disrupt the rote learning loop of verbatim repetition and reproduction of concepts. The system is envisioned and developed for CS majors in Pakistan and is situated in their very own context. The paper also report in-situ evaluations of first prototype of the proposed system in real environment with end-user learner. The rest of the report is structured as following. The next section reviews the related work, followed by a discussion on the proposed innovation's architecture. Section 4 details the research methodology used to experiment with the prototype of AEH. The results of the analysis are presented in Sect. 5, with discussion in Sect. 6. Finally, the authors conclude the paper in Sect. 7.

2 Related Work

Liu and Hmelo-Silver [11] report on the design and development of educational hypermedia to promote meaningful learning of complex systems in science students. They argued that conventional learning methodologies for complex systems did nothing more than piling up the information in learners' head. To resolve the issue, two different hypermedia versions were tested with 7th graders and pre-service teachers. The authors found that both type of hypermedia support avoidance of rote learning.

Jacobson and Archodidou developed Knowledge Mediator Framework (KMF) [12]. With a proof-of-concept applied to high school students in learning neo-Darwinian evolutionary biology, they tested the efficacy. Their results showed a significant improvement in students' progress, as well as in learning patterns. They noted that the students started developing expert-like models in their solutions.

Though not in a strict hypermedia sense, Zydney and Grincewicz experimented with a multimedia learning environment with videos to enhance students' meaningful learning abilities [13]. Their study found that the amount of time students spent with the system was a predictor of their performance.

Rum and Ismail [14] used metacognitive tools to assist students in learning programming in meaningful ways. They devised six different strategies, however all metacognitive. The tools were implemented with the help of an educational hypermedia. They enrolled 30 participants in experimental group and 36 in control group. The experimental group exhibited performance improvement over the control group.

3 The Proposed Innovation

The proposed AEH is composed of 5 modules, namely student model, assessment engine, adaptation controller, content store, and the interface. The students communicate with the interface, which presents learning/assessment activities to the learner, formatted and selected by the adaptation controller. The adaptation controller works with the information stored in the student model to appropriately select and format the content which is stored in the content store. The selection of learning activities is based on three criteria items relating to a student, namely cognitive style, learning style, and background knowledge. Moreover, the adaptation controller selects/formats assessment activities based on the input provided by the assessment engine. The schematic of the system is depicted by Fig. 1.

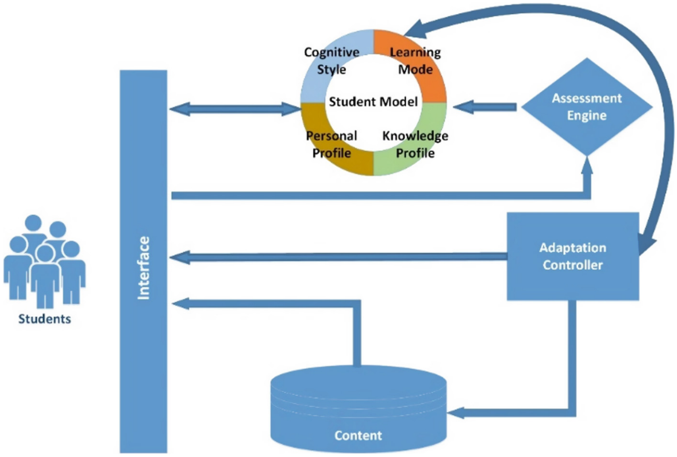


Fig. 1. Schematic of the proposed innovation

Nonetheless, situating every factor in the context of the learner is important for an effective learner model, but some attributes weigh more than others. An important aspect to consider in the case of developing nations is the difference of cognitive style of learners as compared to the ones from developed nations. The authors thus chose Kirton's innovator-adaptor model [15] which is more closely related to rote/meaningful learning than other cognitive models. The authors' stance also finds support in an experiment conducted at the University of Central Punjab (UCP), Pakistan. However the result of that experiment is the topic of another paper [16].

The learning styles model—not to be confused with cognitive styles catering rote/meaningful learning—used in the system is based on VARK (Visual, Auditory, Read-Write, Kinesthetic) model by Fleming and mills [17]. The VARK model is repeatedly reported to be found in learners from developing nations [18]. The authors, however, could not implement kinesthetic style due to software limitations.

Finally, the knowledge profile stores information about the current progress of the learner, upon which a new activity—learning or assessment—is selected for presentation. For further details of the system, the interested reader is referred to [16].

3.1 Disrupting the Rote Learning Loop

The system deploys a novel mechanism to disrupt the rote learning loop. The schematic of the proposed design is given in Fig. 2. Upon starting a learning session, the system —based on learner's model—selects and presents a particular lesson—say Γ—to learner. A learning lesson typically comprises 9 learning activities—3 visual, 3 auditory, and 3 reading, denoted Γv, Γa, Γr, respectively—all focused on the learning theme of that particular lesson. The arrangement of the presentation is decided according to learner's attributes. One learner may get visual first, then auditory, and then read, while another may receive auditory-visual-read, or any other combination.

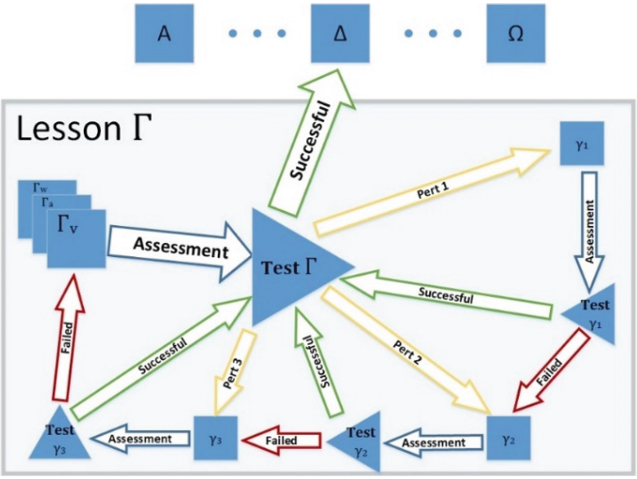


Fig. 2. The learning loop disrupting the rote memorization

After completing the lesson Γ, the learner takes assessment "Test Γ". The tests are designed in a way that they track down the perturbations back to the segment of knowledge where the misconception is stemming from. For example, consider a question on loops in C++. The question may have 4 answers, amongst which one is correct. The rest three are distractors, designed in a way that they point into the direction where the learner's knowledge is erroneous. For example, answer *b* tells that learner has a problem with understanding conditional statements and relational operator, and option *c* may reveal that the learner is not good with pre/post increment concept. Designing the assessment in such fashion allows tracking the source of perturbations.

If a learner is successful in assessment, she moves to next segment of knowledge taking the shortest possible path within knowledge domain. However, if the assessment is incorrect, or partially correct, she may take one of several possible paths. A major error in learning of lesson Γ takes her to sub-activity $\gamma 1$, a lesser problem to $\gamma 2$, and a still lesser problem to $\gamma 3$—increasing subscripts denote reducing magnitude of error. If the learner is taken to $\gamma 1$—a sub-activity of Γ presented in a different way to avoid rote learning—she has to take sub-activity assessment "Test $\gamma 1$" as well. Completing $\gamma 1$ successfully moves her back to main Γ. However, if she is not able to successfully complete $\gamma 1$, she is taken one level further down to $\gamma 2$. A successful completion of "Test $\gamma 2$", takes to the main assessment, and failure takes one step further down to $\gamma 3$. If the learner is not able to complete the simplest level $\gamma 3$, she is then taken back to do the entire learning of lesson Γ again.

3.2 Assessment Model

An important aspect in learning is assessment. If learners are expected—or trained—to produce verbatim (principles, rules, formulas, definitions) answers in assessments, they incline towards bare memorization of facts, i.e. rote learning [19]. To incline them towards more meaningful learning experiences, the assessments shall be designed and implemented with different expectations—no verbatim answers expected.

The test to be conducted on the students was divided into two parts: standardized and adaptive. The standardized test had same questions and rubric for all the participants [19] and the scoring was done as (1), where $Q_{\gamma s}$ represents the quiz from lesson γ with standardized questions, r_i represents the response to the i^{th} item in the respective quiz. The response is calculated as (2).

$$Q_{\gamma s} = \sum\nolimits_{i=0}^{n} r_i \tag{1}$$

$$r = \begin{cases} 1, & correct\,answer \\ 0, & incorrect\,answer \end{cases} \tag{2}$$

Therefore, this part of the assessment followed "criterion-referenced score interpretations" scheme [20], which only considers if the students' answer is correct or not. As an outcome of this approach, a student may simply be declared as 'fail' or 'pass' for the respective test if the cumulative score is $\geq 50\%$.

The second part of the assessment consisted of an adaptive approach to rate the learner amongst peers. The question bank consisted of calibrated (criterion: difficulty level) items which had been meticulously designed by pedagogues. The starting point of this test was based on the score obtained in the standardized test. The score of standardized test was stratified into 50–69%, 70–85% and >85%. This allowed the appropriate entry point for the candidate into the adaptive quiz. The scoring of the adaptive part was based on the formula in (3):

$$Q_{\gamma a} = \sum\nolimits_{i=0}^{n} r_i \times \frac{1}{a_i} \times \frac{1}{w_i} \times \frac{1}{t_i} \tag{3}$$

The penalty terms in (3) are a (number of times the student changed answer options before submitting), w (number of times the student attempted the same question), and t (the time taken by the student to answer, 1 in case of predefined time limit (90 s) and 2 in case of more time), defined respectively in (4), (5), and (6).

$$a = \begin{cases} 1, & \textit{first click} \\ k, & k^{th}\ \textit{change of option} \end{cases} \tag{4}$$

$$w = \begin{cases} 1, & \textit{first attempt} \\ k, & k^{th}\ \textit{attempt of the item} \end{cases} \tag{5}$$

$$t = \begin{cases} 1, & \textit{within}\ 90\ \textit{seconds} \\ 1 + \left(\frac{1}{p} \times k\right), & k^{th}\ 30\ \textit{second interval} \end{cases} \tag{6}$$

The value of p can be empirically estimated. We used $p = 10$ for the experiments. For the adaptive part of the quiz, a student can get the right answer in the first click and attempt, in which case the r_i will be the score of the respective i^{th} quiz item. For all other cases, the score $Q_{\gamma a}$ will depend upon the contribution from the penalty terms and, thus, can never be 100%.

4 Research Methodology

The efficacy of first prototype was tested with CS1 students at the UCP. All participants were enrolled into same 5 courses, including CS1, Basic Electronics, English 1, Social Studies, and Logical Thinking. The students were not given the choice to select subjects themselves—UCP freshmen are offered a pre-designed track in 1st semester.

The complete enrollment of 4 semesters (S15, F15, S16, and F16) was inducted into the experiment. In S15, the students were taught with conventional methods. The educational process was watched closely and the results were recorded. Meantime, the content developers created English language content, and the system developers prepared the first prototype. As soon as F15, the system was ready to go under first efficacy testing. The content/system refinement continued in parallel with QA and error correction, resulting in an updated prototype for S16, and a further improved form in F16. The results of all these semesters were subjected to statistical analysis.

The course chosen for the analysis was English 1, since the second language learning is an area which is especially affected with rote learning mechanism. Nonetheless, building vocabulary may be argued to base on a bare-memorization technique, but comprehending information from a passage needs some creative thinking.

4.1 Participants

A total of 1161 students participated in the experiment, of which 82 withdrawn from the course of their studies in English 1, or dropped from the program altogether. Of the rest 1079, 108 belonged to the control group. The remaining 971, were subjected to different level of treatment. The students in F15 had the first version of the system, which was improved for S16, and in a still improved form for F16. Hence, the level of treatment for the subsequent semesters was increasing. Table 1 enlists the total number of students in each group. The students under the head grading are those whose data were included into the analysis. The enrollment of both S15 and S16 was divided into 3 sections, while F15 and F16 had 10 and 11 sections respectively.

Table 1. Number of students enrolled, withdrawn, and continued in each semester

	S15	F15	S16	F16
Sections	3	10	3	11
Graded	108	352	133	486
Withdrawn	30	6	5	41
Total	138	358	138	527

Table 2. Number of sections in each semester with respective enrolment

Sections	S15	F15	S16	F16
A	42	27	46	47
B	38	29	49	46
C	28	36	38	–
D	–	27	–	41
E	–	35	–	50
F	–	40	–	51
G	–	40	–	37
H	–	39	–	35
I	–	40	–	48
J	–	39	–	51
K	–	–	–	46
L	–	–	–	34
Total	108	352	133	486

Since, all students belonged to the same semester of same program, hence they were assumed to have similar profile, including prior knowledge, the skills learned, the courses taken, and the level of studies already achieved. Moreover, they were enrolled in UCP via the same admission process/criteria, passing the same admission test, and fulfilling the same entry requirements, hence ensuring a similar knowledge profile across entire population.

The same team taught all 4 semesters. However, the number of teachers engaged differed for each semester. F15 and F16 being more populous had more teachers engaged into teaching than spring semester, as detailed in Table 2.

4.2 Procedure

Since, the fall semesters normally gets more intake, F15 and F16 had more sections as compared to the spring semester. The section assignment was on first come first serve basis. At any given time, only one section was open. As soon as a student was admitted to the program, she was assigned to the open section. Once, the opened section had received enough enrollments, it was closed, and another section in line was open. For example, the first student was admitted to section A, and all the forthcoming 49 other students were assigned to that very section. Once, the section A had 50 enrollments, it was marked close and section B was opened, and so on. Moreover, the teacher assignment for the sections was not known at the time of student enrollment, hence, no teacher preference bias was induced.

Contrary to the conventional learning procedure in S15, the subsequent semester were mostly automated. The lessons were delivered mostly with the help of AEH, though the teachers taught some portions manually as well. The quizzes and assignments were mostly delivered and assessed and recoded through the AEH interface.

Each instrument had a specific weight in the final grade of the students. For example, the quizzes comprised 15% of the total weight, and so do the assignments. The class participation comprised of 5%, and the presentation was weighed 10%. The mid-term and the final-term was 25% and 30% of the total grade, respectively.

All the instruments were designed in a way that they minimized the chances of producing verbatim answers, even in S15. However, S15 procedure does not have perturbation tracking mechanism providing learning iterations over learning modules. The instruments in each semester were analyzed and improved for further administration into upcoming semesters. The major improvement was introducing answers which were more innovative and creative.

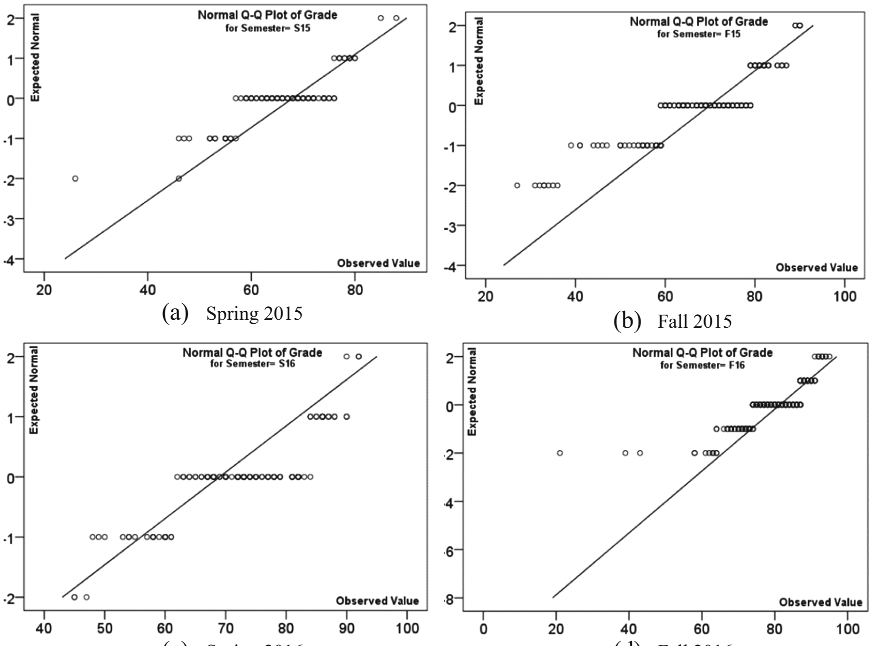
4.3 Tools and Materials

Both manual and automated system included several teaching interventions and assessment instruments designed on similar pedagogical pattern, though differing with respect to technology. Nonetheless, the exact number might have differed in a few cases, but all students of all sections of all semesters received 45 contact hours, either with AEH or without it. On the assessment side, 14 short quizzes were administered on average, on weekly basis—one quiz a week—to track the perturbations in students' current state of knowledge. Similarly, 4 assignments, a class participation activity, a presentation, a mid-term and a final-term exam were administered, either manually or via AEH.

In S15, the students used paper based instruments including all quizzes, assignments, and exams. Nonetheless, the presentation and the class participation activities—and even assignments in some cases—involved the use of multimedia and word processing. Contrarily, almost all tools and materials in F15, S16, and F16 were computer based. All quizzes were administered electronically through the use of AEH, as well as assignments which were delivered and collected through the same platform. However, the exams—mid and final—still remained paper based. One standard outline was followed throughout four semesters.

5 Statistical Analysis and Results

The dependent variable (the numerical grade of students) produced 4 distributions, namely S15 [$N = 108$, $M = 66.93$, $SD = 9.49$], F15 [$N = 352$, $M = 69.71$, $SD = 10.77$], S16 [$N = 133$, $M = 73.26$, $SD = 10.53$], and F16 [$N = 486$, $M = 80.96$, $SD = 7.40$], chronologically representing each semester included into the study. The sample size in each semester varied—the authors had no control over enrollment. To the authors' surprise, no assumptions of parametric analysis was tenable in any distribution. The Shapiro-Wilk test—used to test the normality of data in samples—did not accept the null hypothesis of normality for any distribution [S15: ($p = .004$), F15: ($p < .001$), S16: ($p = .007$), F16: ($p < .001$)]. The same was confirmed through the visual inspection of Q-Q plots as depicted by Fig. 3. Adding to non-normality, large number of outliers appeared in some distributions, as depicted by the boxplot in Fig. 4. The assumption of homoscedasticity was not tenable as well. The Leven's test of homogeneity of variance failed to accept the null hypothesis of equivalent variance in all 4 distributions [$F(3, 1075) = 16.68$, $p < .001$].



(a) Spring 2015

(b) Fall 2015

(c) Spring 2016

(d) Fall 2016

Fig. 3. Q-Q plots of all four distributions

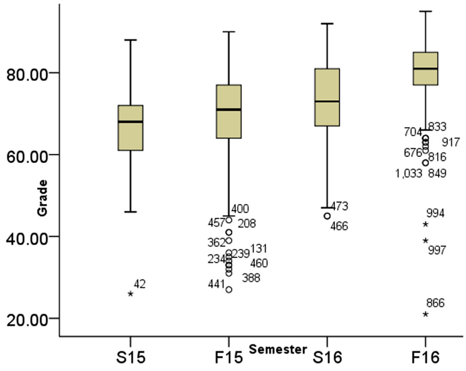


Fig. 4. Boxplot of grade distributions

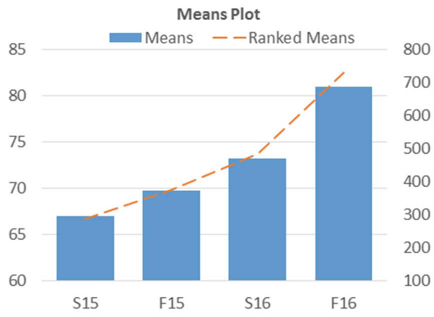


Fig. 5. Means plot comparing arithmetic means of the distributions with ranked means of transformations

Since, the assumptions of parametric analysis were not tenable, the authors opted for Kruskal-Wallis test for Analysis of Variance, reported to be the most favored nonparametric test [21]. The Kruskal-Wallis H statistic showed that there was a statistically significant difference between distributions [$\chi^2(3) = 352.02$, $p < .001$, $R_{S15} = 287.56$, $R_{F15} = 376.78$, $R_{S16} = 483.55$, $R_{F16} = 729.76$]. Further analysis with non-parametric Jonckheere-Terpstra test for ordered alternatives showed that there was a statistically significant trend of increasing median amongst distributions [$J_{JT} = 296705.00$, $z = 18.67$, $p < .001$], remarkably in a chronological order. The same was confirmed by the visual analysis of the means plot presented by Fig. 5. Both means—the means of the original distribution and the means of transformations—had an increasing trend in the chronological order of semesters, i.e. progressing from S15 to F15, and then from S16 to F16, with F16 having the highest mean and median.

6 Discussion and Implications

The statistical analysis of data generated in 4 semesters revealed interesting facts. First, the H-statistic indicated differences in grade distributions of 4 semesters. Then, the means of the raw scores turned out to be rising in chronological order, indicating a positive change in learning, assessed through quizzes, assignments and exams. The positive change was however not attributed to the chance alone, since the data was subjected to chance-corrected statistical methods, like that of Kruskal-Wallis test. The H-test also indicated a difference in the means of transformations created from the original grade distributions. Moreover, the ranking test (Jonckheere-Terpstra) indicated the same chronological ordering, as was observed through visual analysis of means of raw distributions.

The authors draw the most important implication that the rote learning hinders the meaningful learning, and hence performance and creativity of the students in respective area. The argument is backed-up in the literature, as well as finds supports in the experiment reported here. Albeit being a less-respected learning technique, the students are somehow compelled to choose rote learning due to its ability to strip the complexity off the topic. Contrarily, the educators' community wants pupils to learn things in more meaningful ways. Nonetheless, achieving this goal is difficult with conventional pedagogies. The teachers can deliver lectures in novel ways, engage students in meaningful activities, design creative assessments, reward for novelty and innovation, but administering all this with a large number of enrollment seems difficult, if not impossible at all.

One promising solution is the use of AEH systems, which were previously targeted over the customized learning experience. The authors suggest that the AEH—and the learning and assessment modules—shall be designed in a way that they enforce students to refrain from rote learning. In support of their argument, the authors have demonstrated how an AEH system can be designed and implemented to disrupt the rote learning loop.

Nonetheless, controlling all variables in a social science educational setting was not possible. However, the researchers tried to keep the execution all the same across all these years. The only difference induced was in form of educational technology used for learning and assessment. Nonetheless, the students also were changed during each term, however, the induction process for new students remained the same and the students—though changing personally—belonged to the same population.

7 Conclusions and Future Work

AEH can help in changing the preferred learning strategy of the student. One possible course of action is to design AEH with situated learner attributes considering factors which are compelling student body to learn via rote methods at large. Additional to the learning and cognitive attributes, the learning management can, as well, push the students towards rote learning. Learning management inculcating rote learning involves (1) designing and implementing such learning activities which loop on the same thing several times, (2) designing and implementing assessments which are answered with remembered concepts, and (3) rewarding for the verbatim answers. The design of an AEH shall also consider these factors as well. The authors have built such an AEH with a proposed novel assessment system, and the experiments show positive effects on learning.

In the future, the authors want to run further real-time experiments with the system to gather more data on the efficacy. The authors also plan on running two separate subjects, one with the AEH, and the other in a conventional manner to compare the results.

Acknowledgments. We would like to acknowledge the support and guidance of Prof. Ashraf Iqbal, Dean Faculty of IT, University of Central Punjab, Pakistan. We are also grateful to The Punjab Group, Pakistan, for funding this research.

References

1. Chin, C., Brown, D.E.: Learning in science: a comparison of deep and surface approaches. J. Res. Sci. Teach. **37**(2), 109–138 (2000)
2. She, H.-C.: Promoting students' learning of air pressure concepts: the interrelationship of teacher approaches and student learning characteristics. J. Exp. Educ. **74**(1), 29–51 (2005)
3. Lithner, J.: Learning mathematics by creative or imitative reasoning. In: Cho, S. (ed.) Selected Regular Lectures from the 12th International Congress on Mathematical Education, pp. 487–506. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-17187-6_28
4. Mokhtar, A.A., Rawian, R.M., Yahaya, M.F., Abdullah, A., Mohamed, A.R.: Vocabulary learning strategies of adult ESL learners. Engl. Teach. **36**, 133–145 (2017)
5. Kılıç, D., Sağlam, N.: Students' understanding of genetics concepts: the effect of reasoning ability and learning approaches. J. Biol. Educ. **48**(2), 63–70 (2014)
6. Lithner, J.: A research framework for creative and imitative reasoning. Educ. Stud. Math. **67** (3), 255–276 (2008)
7. Hacieminoglu, E.: Elementary school students' attitude toward science and related variables. Int. J. Environ. Sci. Educ. **11**(2), 35–52 (2016)
8. Pell, A.W., Iqbal, H.M., Sohail, S.: Introducing science experiments to rote-learning classes in Pakistani middle schools. Eval. Res. Educ. **23**(3), 191–212 (2010)
9. Sonawat, R., Kothari, M.: Rote learning and meaningful learning in mathematics: perspectives of primary school teachers and learning in children. Indian J. Posit. Psychol. **4**(1), 49–54 (2013)
10. Safdar, M.: Meaningful learning and rote learning in physics: a comparative study in city Jhelum (Pakistan). Middle Eastern Afr. J. Educ. Res. **6**, 60–77 (2013)
11. Liu, L., Hmelo-Silver, C.E.: Promoting complex systems learning through the use of conceptual representations in hypermedia. J. Res. Sci. Teach. **46**(9), 1023–1040 (2009)
12. Jacobson, M.J., Archodidou, A.: The design of hypermedia tools for learning: fostering conceptual change and transfer of complex scientific knowledge. J. Learn. Sci. **9**(2), 145–199 (2000)
13. Zydney, J.M., Grincewicz, A.: The use of video cases in a multimedia learning environment for facilitating high school students' inquiry into a problem from varying perspectives. J. Sci. Educ. Technol. **20**(6), 715–728 (2011)
14. Rum, S.N.M., Ismail, M.A.: Metacognitive support accelerates computer assisted learning for novice programmers. Educ. Technol. Soc. **20**(3), 170–181 (2017)
15. Kirton, M.: Adaptors and innovators: a description and measure. J. Appl. Psychol. **61**(5), 622–629 (1976)
16. Hassan, M.M., Qureshi, A.N.: Situating adaptive educational hypermedia into the local context of developing nations. In: Proceedings of the 2017 2nd International Conference on Communication and Information Systems (ICCIS 2017), Wuhan, China (2017)
17. Fleming, N.D., Mills, C.: Not another inventory, rather a catalyst for reflection. To Improve the Academy, Paper 246. DigitalCommons@University of Nebraska, Lincoln (1992)
18. Saga, Z., Qamar, K., Trali, G.: Learning styles-understanding for learning strategies. Pak. Armed Forces Med. J. **65**(5), 706–709 (2015)
19. Ye, L., Lewis, S.E.: Looking for links: examining student responses in creative exercises for evidence of linking chemistry concepts. Chem. Educ. Res. Pract. **15**, 576–586 (2014)
20. Popham, W.J.: Why standardized tests don't measure educational quality. Educ. Leadersh. **56**(6), 8–15 (1999)
21. Lix, L.M., Keselman, J.C., Keselman, H.J.: Consequences of assumption violations revisited: a quantitative review of alternatives to the one-way analysis of variance "F" Test. Rev. Educ. Res. **66**(4), 579–619 (1996)

Gaze Feedback and Pedagogical Suggestions in Collaborative Learning

Investigation of Explanation Performance on Self's Concept in a Knowledge Integration Task

Yugo Hayashi[1,2](✉)

[1] College of Comprehensive Psychology, Ritsumeikan University,
2-150 Iwakura-cho, Ibaraki, Osaka 567-8570, Japan
y-hayashi@acm.org

[2] Human-Computer Interaction Institute, Carnegie Mellon University,
5000 Forbes Avenue, Pittsburgh, PA 15213, USA

Abstract. Learning by doing, such as when learners give explanations to peer learners in collaborative learning, is known to be an effective strategy for gaining knowledge. This study used two types of facilitation technology in a simple explanation task to experimentally investigate those influence on the performance of understanding self's concept during collaborative explanation activity. Dyads were given a topic about cognitive psychology and were required to use two different theoretical concepts, each of which was provided separately to one or the other of them, and explain the topic to each other. Two types of facilitation were examined: (1) use of a pedagogical conversational agent (PCA) and (2) visual gaze feedback using eye-track sensing. The PCA was expected to enable greater support of task-based activity (task-work) and visual gaze feedback to support learner coordination within the dyads (teamwork). Results show that gaze feedback was effective when there was no PCA, and the PCA was effective when there was no gaze feedback on explaining self's concept. This work provides preliminary implications on designing collaborative learning technologies using tutoring agents and sensing technology.

Keywords: Collaborative learning · Knowledge integration
Pedagogical conversational agents · Eye tracking

1 Introduction

Inspired by Vygotsky's socio-cognitive perspective [27], many socio-constructivism researchers have analyzed group interactions and investigated the characteristics of successful and unsuccessful learners in such practices. Moreover, with the emergence of technological innovations such as sensing technology and the development of automated systems such as conversational agents, there have been attempts to design new tutoring systems that enable greater

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 78–87, 2018.
https://doi.org/10.1007/978-3-319-91464-0_8

support of social learning [12]. In such systems, pedagogical agents can play a role as a social actor, e.g., a teacher. Sensing technology has been used to detect users' mental states and as an awareness tool, to support productive interactions among students in social learning [4]. The present study focused on collaborative learning dyads who engage in a concept explanation task and investigated the relative effectiveness of two technologies for supporting learning performance in such activities. We conducted a factorial analysis to determine the effects of using (1) pedagogical agents and (2) gaze-sensing technology for facilitating awareness of collaborative partners.

1.1 Collaborative Learning and Knowledge Integration

Studies in cognitive science have shown that constructive activities such as self-explanation are a metacognitive strategy effective over a wide range of task domains [1,6]. Furthermore, studies in learning science have shown that collaborative interaction enables learners to develop conceptual understandings [10], conceptual changes [22], and higher-level representations [25]. It is also known that the visualization of a problem from different perspectives can be achieved via explanation activities [26]. Classroom practices based on these notions, called "jigsaw learning" [2], are a known technique for facilitating such cognitive processes by explanation activities conducted in groups. Scenarios in which one is required to consider different perspectives may create opportunities to integrate other knowledge and thus develop a higher, more abstract representation of the content [22].

Although constructive interactions such as explanation activities in collaborative learning between partners having different knowledge are an ideal strategy for gaining new knowledge, there are certain aspects related to learners' cognition and communication that should be considered when designing collaborative tutoring systems. As self-explanation studies have shown, unsuccessful learners fail to develop self-monitoring states [6]. It is important to design tutoring systems that facilitate such metacognitive activity in learners and enable them to generate explanations that refine and expand content and problems. Intelligent tutoring systems (ITSs) have proved effective in facilitating such metacognition in learner–system interactions [9,19]. Recent studies on ITSs have shown the effects of teaching via tutoring systems [5], developing conversational agents that have rich detectors for capturing the learner's state and that generate facilitation prompts [8]. Other studies have investigated the relative effectiveness of various types of facilitation prompts given by agents in self-regulated learning [3]. The present study will define the activity supported by such tutoring systems is termed "task-work".

However, most studies have investigated knowledge development and learners' cognitive states through one-to-one interaction with the tutoring system; the number of studies on learning in multiple parties is relatively small. Additionally, social science studies focusing on psychological outputs in group-based activities have pointed out the disadvantages of multi-party learning [15], such as the difficulty of developing common ground between learners [7]. Because communication plays an important role in collaborative problem solving, we consider communication support to be an important factor in ITS design. We define this as “team-work”.

1.2 Supporting Task-Work and Team-Work in Collaborative Learning

Pedagogical Conversational Agent. The emerging technology for developing pedagogical conversational agents (PCAs) as virtual teachers has become recognized as an effective way to support learners. The use of conversational agents in collaborative problem solving has been shown to be effective in prompting achievement of goals [16], providing periodic initiation opportunities [20], collaboratively setting subgoals together [11] and showing scripted dialogues to learners [23]. Several studies have investigated the influence of a PCA’s functional design on knowledge explanation tasks such as providing emotional feedback [12], using multiple PCAs upon feedback [13], and using gaze gestures during learner–learner interactions [14]. However, these studies found evidence that learners sometimes ignore or misuse the PCA. Other problems include PCAs’ inability to fully support learner coordination during the activity. It is still not clearly understood what kinds of technology may facilitate the learning process when using a PCA.

Visual Gaze and Real-Time Feedback. As mentioned, one of the problems in collaborative learning on a concept explanation task is the hurdle of establishing common ground between the learners. Cooperative tasks such as the speaker’s language expression and the listener’s understanding process require mutual awareness, prompting the development of awareness tools to support interaction among students in computer-supported collaborative learning (CSCL). Recent studies have shown that providing feedback on the visual gaze of collaborative partners using eye-trackers [17], affording an indication of where the other learner may be looking in the same computer screen, can facilitate the achievement of joint attention.

Previous studies in communication [21] suggest that the degree of gaze recurrence in speaker–listener dyads is correlated with collaborative performance such as understanding and establishing common ground, showing that common knowledge grounding positively influences the coordination of visual attention. Several studies have investigated the use of visibly showing a partner’s gaze during a distance computer learning task [17]. Dyads collaborated remotely on a learning task. In one condition, participants were given information about the partner’s eye gaze on the screen; in a control group, they were not. Results showed that real-time mutual gaze perception intervention helped students achieve a higher quality of collaboration. However, these studies only investigated effects on the success of group coordination. It is not fully understood how such technologies can facilitate learning during collaborations in which PCAs are guiding the learners. With this in mind, in this study we took a broader view, focusing on both task-work and team-work simultaneously to see how the two factors may influence collaborative learning performance.

1.3 Aim of This Study

The aim of this study was to investigate the effect of two technologies on tutoring systems in peer collaborative learning. It focused on the effects of (1) prompting metacognitive suggestions using PCAs and (2) enhancing peer learners' awareness by providing their gaze information. This paper documents the effects on the learning performance, especially focusing on the learner's ability to construct a deeper understanding of self's knowledge.

2 Method

Eighty Japanese students, all freshman-year psychology majors, participated in the experiment in exchange for course credit. They are called "learners" and participated in dyads. When participants arrived at the experiment room, the experimenter thanked them for their participation. The experimenter gave instructions for the task, explaining that they would participate in a scientific explanation task in which they would use technical concepts to explain human mental processing. Before the main task, they were given a free-recall test about the concepts in order to ensure that they did not already know the concepts that would be used in the task. Next, they performed the main explanation task for 10 min. Then, they took the post-test, which was another free-recall test. Finally, they were debriefed.

The dyad's goal was to explain a topic in cognitive science (e.g., human information processing in language perception) by using two technical concepts (e.g., "top-down processing," "bottom-up processing"). As in the "jigsaw" method studied in learning science and popularly used in classrooms for knowledge building, we set up a scenario in which the learners did not know each other's concepts. The experimenter separately provided each of the learners with one of the two concepts. Thus, to be able to explain the topic using the two concepts, they needed to exchange their knowledge via explanations.

The first step was for each learner to explain his/her assigned concept to his/her partner. The concept was provided to the learner before the task began, and a brief description of the concept was also shown to him/her throughout the task. On starting the task, the learners were requested to first read the description and then explain its meaning to their partner. Learners were free to ask questions and discuss the assigned concept with their partners. After one learner finished his/her explanation of his/her assigned concept, they switched roles, and the other learner explained his/her concept. Each learner was also instructed that he/she would need to explain his/her partner's concept so they would both be able to explain the topic using the two technical concepts.

2.1 Experimental System

The present study used a redeveloped version of a system designed for previous studies [12–14], which was developed in Java for a server–client network platform. For this study's purposes, the system featured (1) a PCA that provides metacognitive suggestions to facilitate the explanation activities and (2) real-time feedback on the partner's visual gaze (gaze feedback). Each learner sat before a computer display. They were not able to see each other but were able to communicate with each other orally, and they were instructed to look at the display while conversing with each other.

Participants' Screens and Gaze Feedback. To start, the screens simultaneously changed to the displays shown in Fig. 1. The brief explanation of the assigned concept was presented on the monitor of the corresponding learner, and the explanation of the other learner's concept was covered so they could not simply read and proceed as individuals.

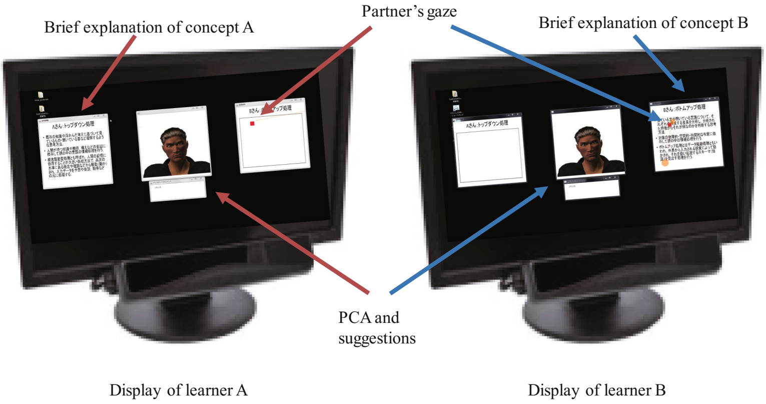


Fig. 1. Example of participants' screens.

The study used two eye-trackers (Tobii X2-30) for gaze feedback; a program was developed to show the visual gaze of the partner during the task as a red square in real time. Since the participants were instructed to begin by reading the text on their screens, it was expected that while one partner (learner B) explained his/her concept by looking at the area with the explanation of the concept, the listener (learner A) would also look at the same area as they proceeded.

PCA. In the center of the screen was an embodied PCA, which included physical movement upon speech, and a text box underneath for displaying messages. The experimenter sat to one side in the experiment room and manually signaled the PCA when to provide the metacognitive suggestions. A signal was issued whenever there was a momentary gap during the dyad's conversation, but no more than one signal was issued within one minute. A rule-based generator determined the type of metacognitive suggestion to offer from among five types based on [12–14].

2.2 Experimental Design

To investigate the effects of the two facilitation methods employed in this study, we implemented a 2 × 2 experimental design (Table 1), each factor representing the absence or presence of the corresponding method (PCA or gaze feedback).

Table 1. Experimental conditions and number of participants assigned to each.

	Without PCA	With PCA
Without gaze feedback	20	20
With gaze feedback	20	20

It was expected that the PCA would be effective for task-work and gaze feedback for awareness of others, thus relating to team-work.

2.3 Data and Analysis

The results to be reported were the effect of the two factors on the dependent variable, which was the gain score derived from the pre- and post-test scores. We coded the data collected by the free-recall pre- and post-tests on explaining the topic of the leaner's self concept. The coding was performed for the explanation of the concept assigned to the learner himself/herself. The following points were given for evaluating the performance on understanding self's concept. (1) 0 points: incorrect, (2) 1 point: naive explanation, but correct, (3) 2 points: concrete explanation based on materials presented, (4) 3 points: concrete explanation based on materials presented and using examples and metacognitive interpretations. The gain scores used for factorial analysis were calculated by subtracting the pre-test scores from the post-test scores.

3 Results

A 2×2 between-subject ANOVA was conducted on the gain score. Figure 2 shows the average gain score for each condition according to the concept explained. There was a significant interaction between the two factors ($F(1, 76) = 4.3563, p < .05, \eta_p^2 = .0542$). Further analysis conducted for the simple main effects shows that the score for the with-gaze-feedback condition was higher than that for the without-gaze-feedback condition when no PCA was used ($F(1, 76) = 7.5622, p < .01, \eta_p^2 = .0905$). Additionally, the score for with-PCA was higher than for without-PCA when learners did not receive visible feedback about their partners' gaze ($F(1, 76) = 9.4563, p < .01, \eta_p^2 = .1107$).

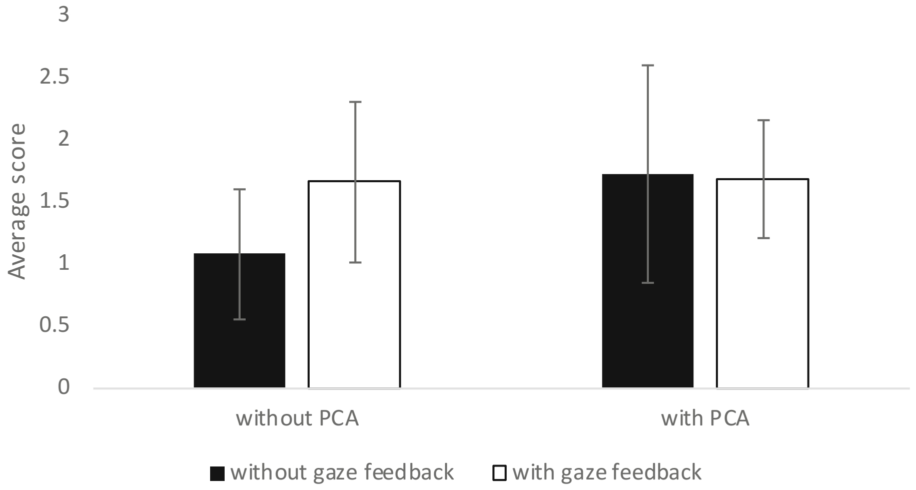


Fig. 2. Results of the free-recall analysis according to the self's concept explained. The error bars represent the SDs.

4 Discussion

The results for the explanation of the self's concept show an interaction between the two factors. The results reveal that the visible gaze feedback was effective when no PCA was presented to the learners. This finding is consistent with those of previous studies [17,24] even though they used other types of tasks and other dependent variables. Although gaze feedback is effective for gaining knowledge through explanation activities, no effect was found for gaze feedback when the PCA was present. The advantage of using the PCA only appeared when there was no gaze feedback on this dependent variable. These results are interesting given the fact that there was no synergistic effect between the two, but there is also no negative influence. Some participants in the with-PCA/with-gaze-feedback condition may have paid attention only to the PCA or to the gaze feedback cues because of their limited attention capability and thus paid less attention to the other system function. Further investigation of how they attended to the PCA and the partner's gaze can provide more details about this point and remains as a future task. Moreover, performance on explanation of the learner can be reanalyzed by using coding methods which focus on coordination and knowledge integration. These are the challenges for the future.

One interesting observation is that the post-test scores were relatively low. The average post-test scores by condition were without-PCA/without-gaze-feedback, 1.12; without-PCA/with-gaze-feedback, 1.68; with-PCA/without-gaze-feedback, 1.92; with-PCA/with-gaze feedback, 1.79. The average score, less than 2, indicates that many learners used naive explanation strategies. This is particularly interesting in light of the fact that such tendency was rapidly to occur when they were giving explanations of their own assigned concept. Such egocentric bias is also seen in collaborative problem-solving tasks in studies in the field of cognitive science, and they are considered to arise during communication [18].

5 Conclusion

In collaborative learning, explaining is known to be an effective strategy for reflecting and gaining knowledge. Incorporating this concept, this study was an experimental investigation using a simple explanation task in which two learners having different knowledge were asked to explain a particular topic in cognitive science. The purpose was to investigate the kinds of technology that might facilitate the learners' gaining of knowledge about learner's own concept. The study focused on two types of technology, each designed to facilitate a different aspect of learning important in collaborative learning, task-work and team-work. The first was the use of a PCA that provided metacognitive suggestion prompts; this was expected to facilitate their task-work of explanation activities. The second was the use of sensing technology showing the partner's gaze location; it was expected that such a visual aid would provide a better opportunity to coordinate and establish common ground. Used together, these two technologies were expected to facilitate learning performance, which was measured by the dependent variable, the level of understanding of self's concepts. The results on performance of explanation about self's concept show that gaze feedback was effective when there was no PCA, and the PCA was effective when there was no gaze feedback. Analysis based on interaction process related to coordination and looking at the performance based on knowledge integration should be challenges for the future work. This work provides preliminary results and contributes to the development and design of collaborative learning technologies using tutoring agents and sensing technology.

Acknowledgments. This work was supported by Grant-in-Aid for Scientific Research (C), Japan Society for the Promotion of Science (JSPS) KAKENHI Grant Number 16K00219.

References

1. Aleven, V.A., Koedinger, K.R.: An effective metacognitive strategy: learning by doing and explaining with a computer-based cognitive tutor. Cogn. Sci. **26**(2), 147–179 (2002)
2. Aronson, E., Patnoe, S.: The Jigsaw Classroom: Building Cooperation in the Classroom, 2nd edn. Addison Wesley Longman, New York (1997)
3. Azevedo, R., Cromley, J.: Does training on selfregulated learning facilitate students' learning with hypermedia? J. Educ. Psychol. **96**(3), 523–535 (2004)
4. Ben Khedher, A., Jraidi, I., Frasson, C.: Assessing learners' reasoning using eye tracking and a sequence alignment method. In: Huang, D.-S., Jo, K.-H., Figueroa-García, J.C. (eds.) ICIC 2017. LNCS, vol. 10362, pp. 47–57. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-63312-1_5

5. Biswas, G., Leelawong, K., Schwartz, D., Vye, N.: Learning by teaching: a new paradigm for educational software. Appl. Artif. Intell. **19**(3), 363–392 (2005)
6. Chi, M., Leeuw, N., Chiu, M., Lavancher, C.: Eliciting self-explanations improves understanding. Cogn. Sci. **18**(3), 439–477 (1994)
7. Clark, H.H., Brennan, S.E.: Grounding in communication. In: Resnick, B.L., Levine, M.R., Teasley, D.S. (eds.) Perspectives on Socially Shared Cognition, pp. 127–149. APA Press (1991)
8. D'Mello, S., Olney, A., Williams, C., Hays, P.: Gaze tutor: a gaze-reactive intelligent tutoring system. Int. J. Hum Comput Stud. **70**(5), 377–398 (2012)
9. Graesser, A., McNamara, D.: Self-regulated learning in learning environments with pedagogical agents that interact in natural language. Educ. Psychol. **45**(4), 234–244 (2010)
10. Greeno, G.J., de Sande, C.: Perspectival understanding of conceptions and conceptual growth in interaction. Educ. Psychol. **42**(1), 9–23 (2007)
11. Harley, J.M., Taub, M., Azevedo, R., Bouchet, F.: "Let's set up some subgoals": understanding human-pedagogical agent collaborations and their implications for learning and prompt and feedback compliance. IEEE Trans. Learn. Technol. **11**(1), 54–66 (2017)
12. Hayashi, Y.: On pedagogical effects of learner-support agents in collaborative interaction. In: Cerri, S.A., Clancey, W.J., Papadourakis, G., Panourgia, K. (eds.) ITS 2012. LNCS, vol. 7315, pp. 22–32. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-30950-2_3
13. Hayashi, Y.: Togetherness: multiple pedagogical conversational agents as companions in collaborative learning. In: Trausan-Matu, S., Boyer, K.E., Crosby, M., Panourgia, K. (eds.) ITS 2014. LNCS, vol. 8474, pp. 114–123. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-07221-0_14
14. Hayashi, Y.: Coordinating knowledge integration with pedagogical agents. In: Micarelli, A., Stamper, J., Panourgia, K. (eds.) ITS 2016. LNCS, vol. 9684, pp. 254–259. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39583-8_26
15. Hill, G.W.: Groups versus individual performance: are N+1 heads better than one? Psychol. Bull. **91**(3), 517–539 (1982)
16. Holmes, J.: Designing agents to support learning by explaining. Comput. Educ. **48**(4), 523–547 (2007)
17. Jermann, P., Mullins, D., Nuessli, M.A., Dillenbourg, P.: Collaborative gaze footprints: correlates of interaction quality. In: Proceedings of CSCL 2011, pp. 184–191 (2011)
18. Keysar, B., Barr, J.D., Balin, A.J., Brauner, S.J.: Taking perspective in conversation: the role of mutual knowledge in comprehension. Psychol. Sci. **11**(1), 32–38 (2000)
19. Koedinger, K.R., Anderson, J.R., Hadley, W.H., Mark, M.A.: Intelligent tutoring goes to school in the big city. Int. J. Artif. Intell. Educ. (IJAIED) **8**, 30–43 (1997)
20. Kumar, R., Rose, C.: Architecture for building conversational architecture for building conversational agents that support collaborative learning. IEEE Trans. Learn. Technol. **4**(1), 21–34 (2011)
21. Richardson, D.C., Dale, R.: Looking to understand: the coupling between speakers' and listeners'eye movements and its relationship to discourse comprehension. Cogn. Sci. **29**(6), 1045–1060 (2005)
22. Roschelle, J.: Learning by collaborating: convergent conceptual change. J. Learn. Sci. **2**(3), 235–276 (1992)

23. Rummel, N., Spada, H., Hauser, S.: Learning to collaborate while being scripted or by observing a model. Int. J. Comput.-Supported Collab. Learn. **4**(1), 69–92 (2009)
24. Schneider, B., Pea, R.: Toward collaboration sensing. Int. J. Comput.-Supported Collab. Learn. **4**(9), 5–17 (2014)
25. Schwartz, L.D.: The emergence of abstract representation in dyad problem solving. J. Learn. Sci. **4**, 321–354 (1995)
26. Shirouzu, H., Miyake, N., Masukawa, H.: Cognitively active externalization for situated reflection. Cogn. Sci. **26**(4), 469–501 (2002)
27. Vygotsky, L.S.: The Development of Higher Psychological Processes. Harverd University Press, Cambridge (1980)

Motivational Assessment Tool (MAT): Enabling Personalized Learning to Enhance Motivation

Elizabeth Lameier[1](✉), Lauren Reinerman-Jones[1,2,3](✉), Gerald Matthews[1,2,3](✉), Elizabeth Biddle[2](✉), and Michael Boyce[3](✉)

[1] University of Central Florida, Orlando, FL 32826, USA
{elameier,lreinerman,gmatthews}@ist.ucf.edu
[2] The Boeing Company, Orlando, FL 32826, USA
elizabeth.m.biddle@boeing.com
[3] Army Research Laboratory, West Point, NY 10996, USA
michael.w.boyce11.civ@mail.mil

Abstract. Motivation is a key factor for learning and retention. Motivation in learning, which refers to an individual's desire to learn, is influenced by a number of factors (e.g., interest, self-regulation abilities, self-efficacy, personality) and is further complicated by an individual's sensitivity to those factors. Thus, identifying a learner's general and fine-grained motivation factors is essential to designing individualized adaptations or interventions for implementation in an Intelligent Tutoring System (ITS). The present study addressed the development and validation of the Motivational Assessment Tool to identify correlations between motivation variables and factors from education and psychology. The results indicate an overlap between the scales, which implies a higher-order dimension structure not captured by existing instruments, enabling instructional designers to use the MAT to evaluate the motivation support provided by an ITS overall and identify motivation needs for individual learners.

Keywords: Motivation · Learner's motivational attitude towards learning
Motivation Assessment Tool

1 Introduction

Demands for individualized adaptations are increasing in a range of learning contexts including intelligent tutoring systems (ITSs). The advantages of ITSs in enhancing learning are attributed to the individualized support provided for skill acquisition via feedback directly geared to the student's response and through tailoring learning to the student's existing skills [1]. Recent research has also highlighted the relevance of a broader range of personal characteristics in supporting adaptation to ITS as a learning environment, including motivation, emotion, and self-regulation [1].

Motivation is important to consider when designing instruction within an ITS for two reasons. First, ITS-enabled learning is often delivered with no human instructor present to monitor motivation and provide encouragement or other reinforcement if

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 88–98, 2018.
https://doi.org/10.1007/978-3-319-91464-0_9

motivation is failing. Indeed, in some individuals, an ITS may elicit maladaptive motivation and cognitions due to students misusing the system [2, 3]. Second, features of ITSs, such as immediate feedback and appropriate levels of difficulty, may provide greater motivation than conventional instruction for some learners. Overall, differences exist in learner motivation elicited in the traditional classrooms and in ITS contexts, as well as individual differences in learner motivation across contexts.

It is thus important for the ITS community to develop methodologies for evaluating the motivational qualities of ITSs, as well as determining areas of strength and weakness in individual ITS users. Assessment of the individual's motivation variables will determine their initial orientation to the ITS prior to learning tasks and enable prediction of how their motivation may change in response to positive and negative experiences with the tutor during learning. Understanding motivation factors may allow ITSs to be designed to support motivation in individuals of differing characteristics.

Individual differences in motivation in relation to ITS have been neglected, but some studies have been conducted focusing on achievement goals [4, 5]. The Generalized Intelligent Framework for Tutoring (GIFT) is a service-oriented architecture used to address ITS's limitations such as adapting instruction style and strategy to learner needs [6]. The goal for the Motivational Assessment Tool (MAT; 7) is to capture an individual's general motivation type and motivator preferences that could be implemented as ITS adaptations. Rather than solely relying on a learner's cognitive ability, the MAT establishes an additional trait-based relationship with the learner. For example, a learner that the MAT classifies as learning driven (e.g., possessing higher intrinsic motivation tendencies) would be provided higher-levels of autonomy (e.g., choices) in how they execute a task or training. If a learner is categorized as prone to high levels of stress and low competency, perhaps correlated with neuroticism and less intrinsic motivation tendencies, they could benefit from a more structured course that focuses on frequent positive feedback, sensitivity to mistakes, additional attempts, more opportunities to feel success, and longer periods of guidance. Moreover, an ITS can also adapt to specific motivators facilitating a higher level of motivation when learning. Learners that have a loss of effort while learning, but are not emotionally sensitive, perhaps are motivated by competition, various uses of time pressure, acknowledgements, etc. The goal for this paper is to discuss psychometric analyses of the MAT and its implications for ITSs. The second goal is to evaluate learner's motivational preferences for ITSs and the classroom to find motivational possibilities and limitations for incorporation in an ITS.

1.1 Motivational Factors in ITS

Motivation is a key factor for human performance and learning retention [8], and originates from multiple sources [e.g. 7]. The challenge for assessment is that the individual's learning experience reflects a complex web of interacting motivation variables, which together support adaptation to the learning environment. One approach to identify the composition of a learner's motivation is to utilize a variety of existing motivation constructs and measures that have been validated in traditional and online settings and seem relevant to ITS, which are sought to be delivered online (e.g. cloud).

The Achievement Goal Theory is a main theory of motivation in educational research [9]. The 3 × 2 model crosses valence (approach or avoidance) with self-goals (past experiences), task-goals (e.g., current experiences, understanding and answering questions correctly), and other-goals (comparison to others) [9]. Traditional and online learning environments suggest that mastery goals are typically more adaptive than performance goals. Mastery learners were more likely to take notes and seek information within a technology-enhanced learning environment that provided tools for autonomous learning. Two studies of ITS [5, 10] looked at responses to scaffolding provided by the tutor, i.e., prompts and feedback based on the learner's behavior. Both found that performance-approach learners benefited more from scaffolding than mastery-approach learners. Scaffolding may undermine mastery learners' sense of challenge, but motivate performance learners to compete with their peers [5].

Another widely used questionnaire for student motivation is the Motivated Strategies for Learning Questionnaire [MSLQ; 11]. It assesses four components of motivational orientations and learning strategies, based on a theoretical account of self-regulation [11]. It was developed for a classroom environment. The MSLQ is important to an ITS because it correlates the learner's motivation with the learner's final grade [12]. This comparison allows the ITS to predict the learners that will score higher in a course.

Grit describes a trait of a person's perseverance and passion for long-term goals [13]. Learners higher in grit are better able to handle challenges as they work towards long-term goals, leading to higher achievement, engagement and retention [14]. Grit is potentially an important factor to consider for an online learning or ITS setting. On the one hand, features of ITSs, such as regulation of difficulty of assignments and adaptive provision of scaffolding, may reduce or enhance the role of grit. ITSs should remove obstacles to learning. However, individuals dependent on the social reinforcement of the classroom may require grit to maintain motivation online.

Much educational research draws a broad distinction between deep and surface learning [15]. Deep learners are able to understand the material and surface learning with memorization and rote learning. The deep-surface distinction fuses cognitive and motivational elements of learning: deep learners are motivated to understand whereas surface learners are motivated to reproduce material [16]. The Revised Study Process scale (R-SPQ-F2) is commonly used to measure both tendencies [17]. ITSs could adapt to motivate learners that have deep vs. surface level learning tendencies. A deep-learner version might provide links to material that the learner could explore autonomously, whereas a surface-learner version would test and provide feedback especially on rote reproduction of material (if pedagogically acceptable).

The instruments reviewed have had substantial educational impacts and may contribute to investigating the role of motivation in ITS. However, it is unclear whether they provide comprehensive assessment of relevant factors. Therefore, the MAT [7] is being developed to assess a comprehensive, individualized learner profile of motivation. The MAT aims to inform assessment and adaptions of learning, e.g., by comparing the motivation variables typically elicited by classroom and online settings. It also aims to specify the factors motivating the individual in a range of environments including ITSs, as well as traditional classroom and online settings.

1.2 Motivation Assessment Tool (MAT)

The first iteration of the MAT was based on a compilation of 31 previously published motivation assessment instruments [7] that were clustered by similarities and reduced to basic constructs. Additional items were also developed to evaluate the types of reinforcers that will help support an individual's motivation. The second iteration, used in this study, added scales important to motivation that did not cluster in the first iteration, such as autonomy, Table 1. The MAT was delivered through GIFT [6, 18]: see [19].

Table 1. Scales in the MAT

General scales	Learning driven, autonomy, goal orientation, loss of effort, worry, competition, positive outlook, support preference, self-regulation, workload, challenge, organize and structure, social, effort based on punishment, frequency/extinction, relatedness, anxiety, freeze, and fear scale, breaks
Motivator inventory scales	Digital, energizer, logical consequences, low value, high value, self-reward, activities, hobbies, feedback, acknowledgement, level of interactive media instruction (imi), time during learning, time after learning, sensors

The first goal for present study was to identify higher-order motivational factors from the MAT and select existing motivational scales using exploratory factor analysis. Specification of motivation factors supports the longer-term aim of developing a multi-stratum structural model for learner motivation that distinguishes broad (general motivation) and narrow (specific reinforcers) aspects of motivation. The second goal for the present study was to compare the MAT with existing scales as a predictive tool of preference for ITS learning environments. Accomplishing those entailed developing a scale for preference, followed by multivariate analyses to compare broad motivation factors with more-narrowly defined constructs as predictors of motivator preference.

2 Method

2.1 Participants

201 participants (89 females, 112 males) were recruited through Amazon Mechanical Turk, with ages ranging from 22 to 62 years.

2.2 Materials and Procedures

The refined MAT in this study comprised 485-items. The MAT was compared to several motivational assessments: MSLQ, Short Grit and Ambition Scale, 3 × 2 Goal Orientation Questionnaire, and the Revised Study Process Questionnaire. The MSLQ, is one of the most extensively used questionnaires for motivation, with 31 questions assessing motivation and 50 questions assessing strategies, [11]. The Short Grit and Ambition Scale measured grit and ambition, with 17-items [13]. The 18-item 3 × 2 Goal

Orientation Questionnaire has 6 achievement goals describing a learner's goal in an academic setting, [9]. Finally, the Revised Study Process Questionnaire, [17], is a 20-item questionnaire on learners' approaches to studying (deep or surface). Using an online interface for GIFT [18], participants completed a demographics questionnaire, read instructions for each questionnaire, and answered all survey questions.

3 Results

3.1 Psychometric Properties of Motivational Scales

Cronbach alpha coefficients for the MAT scales were generally acceptable and similar to those found in the initial study of the instrument [19]. For the 15 MAT scales, the median alpha was 0.86 (range: .581–.951). Scales with alphas < .70, and thus in need of further refinement, were Breaks (.581) and Support preference (.680). Alphas for the additional motivational scales were also acceptable (range: .52–.93), although 3 of the MSLQ failed to reach the conventional standard of .70.

To assess convergence of the MAT with existing motivation scales, an exploratory factor analysis was run, including the 15 MAT scales and subscales from the MSLQ (15 subscales), the Grit Scale (2), the Goal Orientation scale (6), and the Revised Study Process Questionnaire (2). A principal factor method was used for factor extraction. Based on the scree test and parallel analysis [20], four factors were extracted, explaining 63% of the variance. Factors were rotated using the direct oblimin method, which allows factors to correlate to improve factor structure. Factor correlations did not exceed .38 (Factor 1 vs. Factor 3).

Table 2 shows major loadings (>.40) from the factor pattern matrix, which corrects loadings for factor intercorrelation. Variables typically loaded on a single factor only, with a few exceptions.

The first factor was labeled Self-directed Motivation, as it is defined by high levels of self-regulation and associated strategies (including deep learning motives), high self-efficacy, as well as autonomy and positive outlook. Students with these attributes are likely to succeed in both online and classroom settings, although structured and constrained assessments may lessen their motivation. The second factor was labeled Threat Vulnerability as major positive loadings include various scales related to stress and anxiety, as well as needs for breaks and difficulties with sustaining effort. Conversely, negative loadings include MSLQ effort regulation and grit. This factor contrasts the student likely to become demotivated and perform poorly under stress, with the more resilient learner able to overcome obstacles and negative emotions. Factor 3 was labeled Achievement Goals because it is defined by all of the Goal Orientation subscales, as well as related MAT scales including Competitiveness. Avoidance motives for achievement tend to predominate over approach motives. The factor may identify the learner especially motivated by needs to avoid falling short of performance standards; frequent feedback related to performance targets from online instruction might engage such motives. Factor 4 was a smaller factor labeled Social Resources as it is defined primarily by two MSLQ scales that refer to needs to work with others.

Table 2. Summary of factor pattern matrix: highest loadings on motivational scales for each factor.

Higher order MAT factors	MAT scales	MSLQ, goal orientation, study process, grit
1. Self-directed Learning	Self-regulation (.790), autonomy (.777), positive outlook (.762), learning driven (.709), organized structure (.676)	Elaboration (.661), metacognitive self-regulation (.651), task value (.604), rehearsal (.582), intrinsic goal orientation (.574), organization (.544), control belief (.524), self-efficacy for learning and performance (.523), time and study environment (.508)
2. Threat Vulnerability	Workload (.877), support preference (.872), Anxiety/freeze/fear (.878), worry (.828), breaks (.758), loss of effort (.704)	Effort regulation (−.668), test anxiety (.650), grit (−.601), time and study environment (−.557), surface approach (.530)
3.Achievement Goals	Competitiveness (.563), social link (−.477), goal orientation (.476)	Other-avoidance (.865), self-avoidance (.672), task avoidance(.635), other-approach (.643), self-approach (.473), task approach (.466), extrinsic goal orientation (.546),
4. Social Resources		Peer learning (.787), help seeking (.677), deep approach (.532)

3.2 Predictors of Preference for Classroom vs. Online Environments

The scale for attitudes towards classroom vs. online environments comprised 22 features of motivation that might be more characteristic of one or other environment from a 5-point Likert scale (Classroom: Strongly agree vs Online: Strongly agree). Table 3 orders these features by mean score. The scale midpoint is 3, so scores below this value indicate that the feature applies more to the online environment, whereas high scores link the feature to classroom instruction. One-sample *t*-tests, with the Bonferroni correction applied, were run to test which means differed significantly from 3. This analysis identified various distinctive characteristics of the two environments, in line with expectation. Online instruction is better at supporting time management. The social interaction afforded by the classroom appears to enhance both competitive and collaborative motivations, but also tends to elevate stress and anxiety. The classroom is also perceived as providing more personalized feedback.

Table 3. Means and SDs of ratings for features of online and classroom environments, ordered by means. One-sample *t*-test statistics compare mean rating to scale midpoint of 3.

Environment feature	Mean (SD)	*t*
Opportunity to take breaks as needed	2.24 (1.70)	−7.33**
More time to balance learning and personal life	2.40 (1.55)	−5.52**
+Less stressful	2.49 (1.46)	−4.49**
Less feelings of overload from information	2.76 (1.48)	−2.29
Safer for learning complex tasks	2.95 (1.47)	−.53
+Supports remembering information	3.16 (1.36)	1.62
+Builds confidence on test and assignments	3.19 (1.36)	1.98
Too challenging for learning	3.19 (1.15)	2.34
+Provides intrinsic interest	3.19 (1.42)	1.94
Better incentives for performance (e.g., points)	3.20 (1.35)	2.09
+Prompts time management and planning	3.30 (1.42)	2.99
+Provides strategies for memorizing (e.g., notes)	3.32 (1.35)	3.35*
+Helps learner get the best test score	3.29 (1.35)	3.02
Supports comfort about participating with peers	3.32 (1.46)	3.11*
+Helps focus when material is boring or difficult	3.41 (1.45)	3.95*
Provides personalized feedback	3.55 (1.31)	5.89**
+Provides motivation to learn with other peers	3.60 (1.35)	6.22**
Satisfies needs for competition	3.63 (1.29)	6.85**
+Encourages application of effort to learning	3.73 (1.37)	3.79*
Effective use of negative feedback	3.78 (1.11)	9.91**
Causes more upset if an answer is wrong	3.90 (1.16)	10.90**
Elevates fear and anxiety	3.98 (1.13)	12.31**

Note. *$p < .05$. **$p < .01$

Analysis demonstrated an underlying general factor of preference for one or the other environment. Items marked with a plus in the left column of the table showed the highest loadings on a general preference factor. A scale was constructed from these 10 items (alpha = .913). High scorers tended, for example, to see the classroom environment as more supportive of effort and focus, more interesting, and relatively less stressful than the online environment.

A multivariate approach was taken towards identifying predictors of the person's preference for classroom vs. online learning. Relationships were assessed between the four higher-order factors described previously and preference, using a multiple regression model. Partial correlations were used to investigate whether individual motivational scales predicted preference, over and above the four factors. Based on the partial correlations, regression was conducted incorporating selected individual scales.

Factor scores for the factor model shown in Table 4 were computed using the regression method. The initial multiple regression, including the four motivation factors as predictors of preference, was significant, $R = .42$, $F(4, 194) = 3.63$, $p < .01$. Next, we tested whether individual motivation scales added to the predictive power of the four broad factors by computing the partial correlation between each scale and the preference measure, controlling for the four factors. To reduce Type 1 error, and to exclude small-magnitude associations, a significance level of $p < .01$ was applied to these analyses. For the MAT scales, two partial correlations reached significance. Preference for the classroom environment was associated with lower levels of autonomy ($r_p = -.28, p < .001$) and with loss of effort ($r_p = -.28$, $p < .001$). For the additional motivational scales, none of the partial correlations attained significance.

Table 4. Summary statistics for regression of preference for learning on motivational factors.

Predictor	B	*r*
F1: Self-Direction	−.24**	−.15*
F2: Threat Vulnerability	.07	.15*
F3: Achievement Goals	.11	.05
F4: Social Resources	.16*	.10
Residual: Autonomy	−.25***	−.27***
Residual: Loss of Effort	.18**	.21**

Note. * $p < .05$, ** $p < .01$, *** $p < .001$

The partial correlations suggested that variance unique to two of the MAT scales enhances prediction of preference. However, factors and their defining variables cannot be included in the same regression equation because of the likelihood of collinearity. Thus, we regressed the two MAT variables of interest against the four factors, and computed the residual variance unique to the variable. We then ran a two-step hierarchical regression, entering the four factors at step 1, followed by the residuals for the two MAT scales at step 2. At step 2, the two residuals added an additional 10% to the variance explained, a significant increment in R^2 ($p < .001$). The final equation was significant, $R = .42$, $F(6, 192) = 6.72$, $p < .001$. Checks for collinearity of predictors did not reveal any issues (variance inflation factors were less than 2 for all).

Table 4 summarizes the final equation, following step 2, together with the bivariate correlations for predictors. The regression accounts for covariance of the factors, providing a somewhat different picture of predictors to that afforded by the bivariate correlations in isolation. Although Autonomy loaded on the Self-Direction factor, its unique variance adds to prediction: high autonomy was associated with preference for online. Similarly, loss of effort loaded on Threat Vulnerability, but it was specifically difficulties in sustaining effort that predicted preference for classroom learning.

4 Discussion

In this study, the factor structure of motivation was examined using exploratory factor analysis to examine the convergence of dimensions from the new MAT with those of existing motivation scales. The factor analysis indicated considerable overlap across different scales implying a higher-order dimensional structure that is not well captured by existing instruments. At the same time, the more granular assessment of individual motivation dimension explains a substantial part of scale variance beyond the higher-order factor structure. These data support further efforts to develop a multi-stratum model of motivational factors that can be used to guide research and educational practice in traditional and computer-mediated learning environments respectively. In regard to adaptation of the individual's motivational profile in terms of general factors and more fine-grained attributes supports a balanced picture of learner characteristics, and how to regulate them via reinforcement. Future ITSs may incorporate personalized learning tools on the basis of motivational profile. Such tools may allow the ITS to optimize the type of feedback, level of support, tailored learning strategies, level of personal preference needed, sensitivities to reward and punishment, specific motivators, initial level of challenge, and frequency of support [7].

The refinement of motivation assessments to support these objectives is still a work in progress. The impact of the MAT to future ITS implementation is a proactive individualized trait-based motivation assessment tool and framework to support a learner's needs further than cognition. The MAT aims to specify broad factors that underpin a variety of leading motivational scales, as well as narrower dimensions such as autonomy that may be especially important for ITS. It is intended that future versions of the MAT will distinguish the learner's motivation traits or general dispositions from their state motivation in a specific learning scenario or use of specific motivator. Such a development would allow tracking and interpretation of motivation during learning. For example, if a student with high trait Self-Direction showed steadily declining state Self-Direction as she utilized an ITS over time, we could identify an issue with the design or personalization of the tutor.

At the granular level, different sets of dimensions may be important. For example, the utility of the Goal Orientation scale for personalizing scaffolding has already been demonstrated [5, 10], but in the present study neither the Achievement Goals factors nor the residuals for the eight individual goal orientation scales predicted preference for online learning. A future ITS might be able to determine which assessments were appropriate for a particular learning context.

The present study also illustrated the utility of a multi-stratum approach for understanding motivation factors in online learning. Many individuals still prefer the classroom environment, whose social interactions may support motivation and teacher feedback. By contrast, learners typically report more scope for time management and less stress in online environment. Motivational factors predict these preferences. Individuals high in self-direction (at the factor level) and autonomy motivation (at the scale level) favor online learning. Those prone to lose effort over time prefer the classroom environment, presumably because social stimulation helps to restore flagging motivation.

Findings have implications for design of learning environments and adapting student instruction. We identified potential shortcomings of online environments, whether traditional or online. The challenge for ITS designers is to counter limitations of the online format, especially lack of socially-mediated motivation. For example, the ITS could be designed to allow the learner to interact with others, fostering a sense of a learning community. By contrast, ITS design can also capitalize on perceived strengths of online learning, by affording learners management of their own time.

In conclusion, ITSs have the capability to deliver adaptive, personalized instruction that supports enhanced learning and retention relative to the classroom [1]. Realizing this capability requires a deep understanding of learner motivation to support strategic regulation of reinforcement, feedback, and online delivery of assignments. Adapting instruction to the complexity of individual motivation is perhaps the key to optimizing instruction for all learners regardless of their personal characteristics.

References

1. Sottilare, R.A.: Adaptive Intelligent Tutoring System (ITS) research in support of the Army Learning Model—research outline. US Army Research Laboratory (ARL-SR-0284) (2013)
2. Grant, A.M., Campbell, E.M., Chen, G., Cottone, K., Lapedis, D., Lee, K.: Impact and the art of motivation maintenance: the effects of contact with beneficiaries on persistence behavior. Organ. Behav. Hum. Decis. Processes **103**(1), 53–67 (2007)
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.) Intelligent Tutoring Systems, pp. 531–540. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-30139-4_50
4. Azevedo, R., Moos, D., Johnson, A., Chauncey, A.: Measuring cognitive and metacognitive regulatory processes used during hypermedia learning: issues and challenges. Educ. Psychol. **45**, 210–223 (2010)
5. Duffy, M.C., Azevedo, R.: Motivation matters: interactions between achievement goals and agent scaffolding for self-regulated learning within an intelligent tutoring system. Comput. Hum. Behav. **52**, 338–348 (2015)
6. Sottilare, R.A., Holden, H.K.: Motivations for a generalized intelligent framework for tutoring (GIFT) for authoring, instruction and analysis. In: AIED 2013 Workshops Proceedings, vol. 7, p. 1, July 2013
7. Reinerman-Jones, L., Lameier, E., Biddle, E., Boyce, M.: Informing the long-term learner model: motivating the adult learner (Phase 1). Technical report (2017)
8. Ryan, R.M., Deci, E.L.: Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. Am. Psychol. **55**(1), 68 (2000)
9. Elliot, A.J., Murayama, K., Pekrun, R.: A 3 × 2 achievement goal model. J. Educ. Psychol. **103**(3), 632 (2011)
10. Carr (nee Harris), A., Luckin, R., Yuill, N., Avramides, K.: How mastery and performance goals influence learners' metacognitive help-seeking behaviours when using Ecolab II. In: Azevedo, R., Aleven, V. (eds.) International Handbook of Metacognition and Learning Technologies. SIHE, vol. 28, pp. 659–668. Springer, New York (2013). https://doi.org/10.1007/978-1-4419-5546-3_43
11. Pintrich, P.R.: A manual for the use of the Motivated Strategies for Learning Questionnaire (MSLQ) (1991)
12. Pintrich, P.R., Smith, D.A., Garcia, T., McKeachie, W.J.: Reliability and predictive validity of the Motivated Strategies for Learning Questionnaire (MSLQ). Educ. Psychol. Meas. **53**(3), 801–813 (1993)
13. Duckworth, A.L., Quinn, P.D.: Development and validation of the Short Grit Scale (GRIT–S). J. Pers. Assess. **91**(2), 166–174 (2009)
14. Maddi, S.R., Matthews, M.D., Kelly, D.R., Villarreal, B., White, M.: The role of hardiness and grit in predicting performance and retention of USMA cadets. Mil. Psychol. **24**(1), 19–28 (2012)
15. Case, J., Marshall, D.: Between deep and surface: procedural approaches to learning in engineering education contexts. Stud. High. Educ. **29**(5), 605–615 (2004)
16. Marton, F., Säljö, R.: Approaches to learning. In: Marton, F., Hounsell, D.J., Entwistle, N.J. (eds.) The Experience of Learning, 2nd edn, pp. 39–58. Scottish Academic, Edinburgh (1997)

17. Biggs, J., Kember, D., Leung, D.Y.: The revised two-factor study process questionnaire: R-SPQ-2F. Br. J. Educ. Psychol. **71**(1), 133–149 (2001)
18. U.S. Army Research Laboratory: GIFT Virtual Open Campus. ARL, 25 January 2018. https://cloud.gifttutoring.org/dashboard/#login. Accessed 25 Jan 2018
19. Lameier, E., Reinerman-Jones, L., Matthews, G., Biddle, E., Boyce, M.: The motivational assessment tool (MAT) development and validation study (in press)
20. Hayton, J.C., Allen, D.G., Scarpello, V.: Factor retention decisions in exploratory factor analysis: a tutorial on parallel analysis. Organ. Res. Methods **7**(2), 191–205 (2004)

The Impact of Multiple Real-Time Scaffolding Experiences on Science Inquiry Practices

Haiying Li[1(✉)], Janice Gobert[1], Rachel Dickler[1], and Raha Moussavi[2]

[1] Rutgers University, New Brunswick, NJ 08901, USA
{haiying.li,janice.gobert,rachel.dickler}@gse.rutgers.edu
[2] Massachusetts Institute of Technology, Cambridge, MA 02139, USA
moussavi@mit.edu

Abstract. Computer-assisted assessment environments, such as intelligent tutoring systems, simulations, and virtual environments are now being designed to measure students' science inquiry practices. Some assessment environments not only evaluate students' inquiry practice competencies, but also provide real-time scaffolding in order to help students learn. The present study aims to examine the impact of real-time scaffolding from an animated, pedagogical agent on students' inquiry performance across a number of practices. Participants were randomly assigned to one of two conditions: receiving scaffolding or no scaffolding. All participants completed three virtual labs: Flower (a general pretest), Phase Change, and Density. Results showed that students who received immediate feedback during assessment performed better on subsequent inquiry tasks. These findings have implications for designers and researchers regarding the benefits of including real-time scaffolding within intelligent assessment systems.

Keywords: Science inquiry · Educational data mining · Real-Time scaffolding

1 Introduction

The Next Generation Science Standards (NGSS) were developed in order to promote high quality science instruction emphasizing the integration of three central dimensions: science inquiry practices, crosscutting concepts, and disciplinary core ideas [1]. For this vision to be realized, valid assessment of these areas is required. The first dimension, science inquiry practices, consists of eight practices that students are expected to engage in and master in grades K through 12. Some practices include: forming testable questions, carrying out experiments, analyzing/interpreting data, warranting claims with evidence [2], and communicating findings. These practices are difficult to capture and assess using traditional forms of assessment [3]. Hands-on science inquiry experiments that can elicit these practices are demanding for teachers to implement in classrooms and are extremely difficult to grade [4] due to teacher-student ratios and lack of rigor on observation-based scoring. Other traditional forms of assessment that involve multiple choice items do not fully capture student competencies at science practices [3]. Additionally, assessments based on open-response items do not fully or accurately capture students' inquiry practice

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 99–109, 2018.
https://doi.org/10.1007/978-3-319-91464-0_10

competencies [5, 6], yielding both false negatives (skilled science learners who cannot articulate what they know in words) and false positives (students who are simply parroting what they have read or heard but do not understand the content or practices, etc.). The challenges involved in effectively and accurately capturing students' science inquiry practice competencies have led to the development of various technological assessment systems, whose goals are to assess students' NGSS competencies. Specifically, researchers have developed assessments using simulations [7], virtual learning environments [8] and intelligent tutoring systems [3] to measure students' science inquiry performance. The designs of these systems vary depending on assumptions held regarding the role of assessments. Some assessment systems have been designed for the sole purpose of evaluating student performance, without providing support or guided feedback to students to promote learning within the system [i.e. 9]. Other researchers [3, 7, 8], however, note the need for and benefits of assessment systems that also promote student learning by providing different forms of scaffolds and feedback.

Scaffolding refers to hints and supports provided to a student as they engage in a task that may otherwise not be possible for that student to complete independently [10]. Students often have difficulty completing science inquiry tasks without guidance [11], which is why it is important to integrate carefully designed scaffolds into science inquiry contexts such as virtual assessments [12]. Scaffolding in online environments may involve providing hints on how to go about completing a task or providing directed feedback based on student performance on the particular task. Several technology-based inquiry assessment systems designed with scaffolded feedback have been found to benefit learning of science content and process skills [3, 7, 8, 13]. For instance, SimScientist evaluates students based on their performance in interactive simulations [7]. Students receive guidance that becomes increasingly informative based on their performance on practices related to conducting experiments and interpreting data. With this system, Quellmaz et al. [7] found that students were deeply engaged and that students, including English Language Learners, demonstrated improved science inquiry performance after completing the assessments. This system, however, covered only two topics and the impact of scaffolding on student performance for specific inquiry practices was not examined.

Another science inquiry system that used scaffolds to support student learning is Co-Lab [8]. Co-Lab virtual environments were designed for four topics related to environmental and physical sciences. The Co-Lab environment provides faded scaffolding, so the system provides gradually less support to students as they become more experienced with the system. Using Co-Lab, van Joolingen et al. [8] indicated that scaffolded support enabled students to engage with increasingly complex models and data. While the scaffolding in Co-Lab was found to be beneficial, it did not address all science inquiry practices and did not provide real-time, directed feedback to students based on their performance.

Inq-ITS, (**Inq**uiry **I**ntelligent **T**utoring **S**ystem; inqits.com), is a web-based intelligent tutoring and assessment system with interactive virtual simulations for NGSS science topics in the areas of life, earth, and physical science [3]. Inq-ITS has a pedagogical agent, Rex, who provides real-time scaffolding to students as they engage in virtual labs. Real-time scaffolding means that feedback is provided immediately following student actions indicating unproductive behavior or lack of skills, as opposed to other kinds of automated feedback that may be provided before or after a student has completed an activity. The real-time scaffolding in Inq-ITS is not faded but is responsive to student performance on a number of science inquiry practices including: forming questions, planning investigations/hypothesizing, conducting experiments, interpreting data, and warranting claims with evidence [13–17]. Prior studies with this system have found the scaffolding to be particularly effective for both students' learning of practices such as hypothesis formation and conducting experiments [14, 15] as well as interpreting data and warranting claims [16, 17]. Researchers have yet to examine, however, the benefits of scaffolding in Inq-ITS across several science topics for multiple practices within the same study. It is important to investigate the influence of scaffolding across topics and practices.

The present study examined the impact of real-time scaffolding on learning of science inquiry practices within Inq-ITS. The following two research questions were addressed:

RQ1: Does students' overall performance across science inquiry practices (i.e. generating a hypothesis, collecting data, interpreting data, *and* warranting a claim) improve in subsequent science topics (i.e. phase change and density) if they receive real-time scaffolding?

RQ2: If so, for which *specific* inquiry practices (i.e. generating a hypothesis, collecting data, interpreting data, *or* warranting a claim) does real-time scaffolding improve performance?

2 Method

2.1 Materials

This study adopted three virtual labs in Inq-ITS. The Flower virtual lab contains three activities with the aim of fostering understanding about petal loss caused by salt or sugar, and about the changes to the color of a flower caused by adding red dye. The flower virtual lab could be considered the most basic Inq-ITS lab due to the minimal prior knowledge needed to successfully complete the lab and the fact that each independent variable in the simulation has only two levels. The Phase Change virtual lab contains four activities and aims to foster understanding about how the boiling point of water is impacted by a series of independent variables, including: different levels of heat (Low, Medium, and High), different amounts of ice (100 g, 200 g, and 300 g), and different sizes of a container (Small, Medium, and Large). The Density virtual lab contains three activities and aims to foster understanding about the relationship between the density of a liquid and the: type of liquid substance (water, oil, and alcohol), amount of liquid (quarter, half, and full), and shape of a container (narrow, square, and wide). Demos of Inq-ITS activities are available on the website (inqits.com).

2.2 Measures for Inquiry Practices

In each activity, participants complete four stages of inquiry practices, each of which was automatically assessed by our system. The first three inquiry stages constitute the investigative portion of the virtual lab. The last stage involves writing a scientific explanation based on the results of the investigation. The present study examines student performance on inquiry practices during the investigative portion of the virtual lab, as described below. Each practice is measured using our patented algorithms [3, 18]. The first stage of the virtual lab is the Hypothesis/planning investigation stage, where students use a widget (dropdown menu) to formulate a hypothesis based on an activity goal. This practice was measured by correct identification of an independent variable (IV) and a dependent variable (DV). In the Collect Data/conducting experiments phase, students use a widget (clickable buttons) to manipulate the independent variables in a simulation while a data table automatically records and assesses their inquiry for this practice. This practice was measured by testing a hypothesis and conducting a controlled experiment. If the variables were nominal, then data collection was also measured according to whether there was a pair of trials that tested two levels of the target nominal IV. During the Analyze Data/interpret data stage, students use a widget (dropdown menu) to state their claim, identify whether or not their claim supports their hypothesis, and select evidence that supports their claim (clickable). This stage consists two practices. One practice is about data interpretation, which is measured by correctly selecting an IV and DV for claim, interpreting the relationship between the IV and DV, and interpreting the hypothesis/claim relationship. Another practice is about warranting the claim, which is measured by the selection of more than one trial to warrant the claim, selection of controlled trials, providing data for the relationship between the IV and DV, and providing data for the hypothesis/claim relationship.

The sub-components of these practices were automatically scored as 0 or 1 point by educational data mining and knowledge engineering techniques based on whether students demonstrated competency or not [3, 18]. Prior studies have demonstrated the detectors' high performance [3, 14, 15]. Students' total score for each practice was calculated by taking the mean across corresponding sub-practice scores, and the overall inquiry score was calculated by taking the mean across all inquiry sub-practice scores.

2.3 Participants and Conditions

48 middle school students in grade 7 were randomly assigned into a scaffolding condition (hereafter called the Rex condition; $N = 24$) or a condition without scaffolding (hereafter called the No Rex condition; $N = 20$). Students were from three different middle schools located in the North Western United States. Two of the middle schools were public (42.3% and 60.0% of students received free and reduced lunch, respectively) and one was an alternative middle school (82.4% of students received free and reduced lunch). All participants completed three Inq-ITS virtual labs during their regular science class time over the course of one month in the order of: Flower, Phase Change, and Density. Students received regular science instruction between the implementation of the labs. The Flower virtual lab was used as a baseline, in which all the participants completed

Table 1. Statistics for time × condition across three virtual labs.

Time	Condition	Mean	*SD*	*95% C.I.*		*F*	η^2	Power
				Lower	*Upper*			
1 (Flower)	No Rex	0.53	0.20	0.45	0.61	0.05	0.001	0.056
	Rex	0.52	0.26	0.45	0.59			
2 (Phase change)	No Rex	0.53	0.28	0.44	0.62	24.41^{***}	0.368	0.998
	Rex	0.83	0.19	0.75	0.91			
3 (Density)	No Rex	0.82	0.26	0.73	0.92	0.09	0.002	0.060
	Rex	0.84	0.21	0.75	0.93			

Note. *** $p < .001$. *SD* = standard deviation. *C.I.* = confidence interval. $N = 44$; $df = 1, 42$.

three activities without any real-time scaffolding from the animated, pedagogical agent, Rex.

Results of a one-way ANOVA for the Flower virtual lab showed that the total inquiry scores were not significantly different between the two conditions (see Flower in Table 1 for details). Results of a one-way MANOVA (four inquiry practices × 2 conditions) for Flower further showed that performance on each inquiry sub-practice in Flower was not significantly different between conditions (see Table 2 for details). These results indicated that students in the two conditions (Rex and No Rex) were not significantly different on their competencies at inquiry before real-time scaffolding was provided in the first virtual lab. Thus, to investigate the impact of scaffolding, students in the Rex condition received scaffolding from Rex in the second (Phase Change) and third (Density) virtual labs only when they did not demonstrate the presence of competency. In the No Rex condition, students never received scaffolding and could progress between the stages of an activity even if they demonstrated poor performance on inquiry sub-practices.

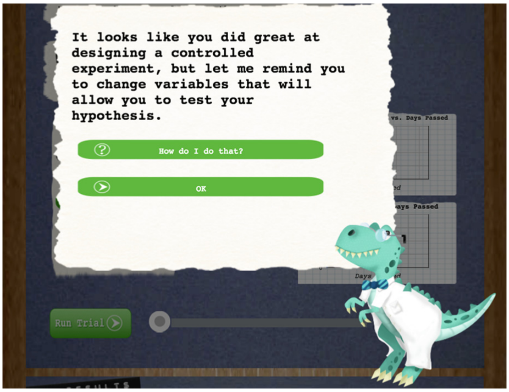
Scaffolding in the Rex condition was provided in real time when the system detected that the student needed support on any of the science inquiry sub-practices. If competency on a particular sub-practice was not demonstrated (for example when the student collects data), Rex would pop-up on the screen with a speech bubble providing a general, orienting hint (see Fig. 1) first. For instance, if a student was running multiple trials in an experiment with the wrong independent variable, Rex would remind the student to look at their hypothesis and make sure they were designing an experiment that tested the hypothesis. The student would click an "okay" button when he/she was finished reading the Rex hint and would then continue with the activity. Some scaffolded Rex hints allow students to request additional information, such as the definition of the term "independent variable" [15, 17].

Table 2. Statistics for practices × time × condition across three virtual labs.

Practices	Time	Condition	Mean	*SD*	*95% C.I.*		*F*	η^2	Power
					Lower	*Upper*			
Generating hypothesis	1	No Rex	0.94	0.16	0.87	1.01	1.88	0.043	0.268
		Rex	0.88	0.16	0.81	0.94			
	2	No Rex	0.83	0.26	0.74	0.91	4.18^{*}	0.090	0.515
		Rex	0.95	0.12	0.87	1.03			
	3	No Rex	0.90	0.17	0.83	0.97	0.26	0.006	0.079
		Rex	0.92	0.13	0.86	0.99			
Collecting data	1	No Rex	0.40	0.30	0.24	0.55	0.11	0.003	0.062
		Rex	0.36	0.38	0.22	0.50			
	2	No Rex	0.50	0.37	0.37	0.63	21.65^{***}	0.340	0.995
		Rex	0.90	0.19	0.79	1.02			
	3	No Rex	0.85	0.22	0.73	0.96	0.17	0.004	0.069
		Rex	0.88	0.27	0.77	0.98			
Interpreting data	1	No Rex	0.62	0.18	0.53	0.70	0.12	0.003	0.063
		Rex	0.60	0.19	0.52	0.67			
	2	No Rex	0.50	0.21	0.41	0.59	16.01^{***}	0.276	0.974
		Rex	0.75	0.21	0.67	0.84			
	3	No Rex	0.79	0.30	0.68	0.91	0.00	0.000	0.050
		Rex	0.79	0.21	0.69	0.90			
Warranting claims	1	No Rex	0.17	0.17	0.06	0.29	0.90	0.021	0.152
		Rex	0.25	0.31	0.14	0.35			
	2	No Rex	0.31	0.31	0.19	0.43	25.05^{***}	0.374	0.998
		Rex	0.71	0.22	0.60	0.82			
	3	No Rex	0.75	0.34	0.62	0.88	0.07	0.002	0.058
		Rex	0.77	0.23	0.65	0.89			

Note. $^{***}p < .001$. *SD* = standard deviation. *C.I.* = confidence interval. $N = 44$; $df = 1, 42$.

If a student demonstrated competency on all sub-practices after receiving scaffolding from Rex, Rex would not appear again. If the students' performance on the sub-practice did not improve with subsequent attempts, then Rex would continue to pop-up and provide hints that gradually became more informative. If other sub-practices were not demonstrated, Rex would provide feedback on them. In other words, Rex would not let students progress from one inquiry stage to the next until they had successfully demonstrated all of the inquiry sub-practices related to the particular stage they were on.



It looks like you did great at designing a controlled experiment, but let me remind you to change variables that will allow you to test your hypothesis.

Fig. 1. Example of a Rex pop-up hint.

3 Analyses, Findings, and Discussion

A repeated measures analysis was performed to investigate whether students' performance on each of the inquiry practices improved with real-time scaffolding provided after completion of the first, baseline virtual lab (i.e. Flower). The two within-subjects factors were the four inquiry practices and time phase of completion (i.e. first, second, or third virtual lab completed). The analyses adopted students' performance on their first attempts before scaffolding was provided by Rex for each inquiry practice, because this performance reflected students' real competency in inquiry practices. The between-subjects factor was the two experimental conditions (Rex versus No Rex). The analyses adopted the mean scores of each inquiry practice across all the activities in each virtual lab.

3.1 Performance on Overall Inquiry Practices Across Time Phases

Results of the repeated measures analysis showed a significant two-way interaction between time and condition, F (2, 41) = 16.36, $p < .001$, $\eta^2 = .444$. Table 1 illustrates the means, standard deviations, and other descriptive statistics. The pairwise comparisons of conditions at each time phase showed that students achieved higher inquiry scores (an aggregated score) in the Rex condition than the No Rex condition in the second virtual lab. There were no significant differences between the two conditions on the third virtual lab. To further explore why there was no difference in performance on the aggregated inquiry score between the two conditions on the third virtual lab, pairwise comparisons of time within each condition were conducted. Results showed that students in the Rex condition achieved significantly higher performance in the second virtual lab relative to the first virtual lab, $p < .001$, Cohen's $d = 1.36$. This increase was not significant from the second to the third virtual lab, but a significant increase was found from the first to the third virtual lab, $p < .001$, $d = 1.35$. In the No Rex condition, no significant increase in performance was found from the first to the second virtual lab, but in the third virtual lab students significantly improved relative to the first ($p < .001$, $d = 1.41$) and second virtual lab ($p < .001$, $d = 1.17$).

These findings indicate that students who received scaffolding from Rex significantly improved their performance on inquiry practices in the second virtual lab versus students who did not receive scaffolding from Rex. However, in the third virtual lab, students who never received Rex's scaffolding caught up with those students who received Rex's scaffolding. These findings imply that whether or not students receive scaffolding, their inquiry performance improved with increased use of Inq-ITS virtual labs; three virtual lab activities (each with 3-4 driving questions), however, are required to yield this change. Additionally, the improvement for students who received scaffolding was much faster than those who did not receive scaffolding.

Therefore, the answer to the first research question is that students' performance on overall inquiry practices greatly improved in the subsequent virtual lab if they received real-time scaffolds. Based on our research design, we believe this improvement is due to Rex's scaffolds that provided students with guidance when they needed it in order to support them in conducting inquiry. The series of scaffolds served to elaborate the reasons why a particular practice was important and the steps involved in successfully engaging in the practice. This form of guided discovery facilitates learning and performance on future inquiry tasks [13–17].

3.2 Performance on Each Inquiry Practice Across Time Phases

As a follow-up to the analyses above, we examined students' performance on each specific inquiry practice of interest (i.e. generating a hypothesis, collecting data, interpreting data, warranting a claim). Results of the repeated measures analysis showed a significant three-way interaction of practices × time × condition, $F\ (6, 37) = 2.53$, $p = .038$, $\eta^2 = .291$. Table 2 illustrates the means and standard deviations of inquiry practices and other statistics. The pairwise comparisons for each inquiry practice showed students achieved higher inquiry practice scores in the Rex condition than the No Rex condition in the second virtual lab. There were no significant differences between the two conditions in the third virtual lab at the specific inquiry practice level; similar to results at the overall inquiry performance level.

The pairwise comparisons showed that students in the Rex condition achieved significantly higher performance in the second virtual lab than in the first virtual lab for practices of data collection, $p < .001$, $d = 1.80$; interpreting data, $p = .004$, $d = 0.75$; and warranting claims, $p < .001$, $d = 1.71$. This increase was not significant from the second to the third virtual lab, but was significant from the first to the third virtual lab for all three practices, i.e., data collection, $p < .001$, $d = 1.58$; interpreting data $p = .002$, $d = 0.95$; and warranting claims, $p < .001$, $d = 1.91$. In the No Rex condition, no significant increase was found from the first to the second virtual lab. However, a significant increase was found from the first virtual lab to third virtual lab for data collection, $p < .001$, $d = 1.71$; data interpretation, $p = .012$, $d = 0.69$; and warranting claims, $p < .001$, $d = 2.16$.

These findings are similar to those for overall inquiry practices, except for the practice of generating a hypothesis, on which students scored very high in the first virtual lab, which suggests that they had already mastered this practice. The answer to the second research question is that real-time scaffolding greatly improved students' performance on data collection, data interpretation, and warranting claims in the second virtual lab. However, students' performance was very similar between the Rex and No Rex conditions in the third virtual lab (i.e. average scores of 0.85–0.88 points for data collection, 0.79 points for interpretation, and 0.75–0.77 points for warranting; see Table 2). This consistent pattern informs us that students' performance on these three inquiry practices does improve with increased use of Inq-ITS virtual labs, but improves faster when real-time scaffolding is provided.

4 Conclusions, Future Directions, and Implications

In this study we investigated whether real-time scaffolds within an inquiry system improved students' inquiry performance; we also investigated the effects of scaffolds on student performance across multiple activities. We found that students who received scaffolding from Rex significantly improved their performance in the second virtual lab relative to those who did not receive scaffolding from Rex on both overall inquiry and on specific inquiry practices. In the third virtual lab, students who never received Rex's scaffolding eventually reached similar levels of performance relative to those who received Rex's scaffolding. These findings imply that whether or not students received scaffolding, their performance eventually improved. Whether this increase in performance, however, was a demonstration of the benefits of discovery learning or the effects of teacher instruction remains unclear. Future studies are needed to further examine the potential impact of in-class instruction that occurs between student's use of virtual labs.

Additionally, students with scaffolding improved their overall inquiry performance as well as performance on the specific practices of collecting data, interpreting data, and warranting a claim faster than those who did not receive real-time scaffolding. Prior studies have explored the benefits of scaffolding in Inq-ITS through modes such as interviews with students [19]. In the future, it would be valuable to attend to whether the number of Rex hints decreased for students from the first to third virtual lab.

Our study provides empirical evidence that a well-designed computer-assisted science inquiry system alone facilitates learning, but adding scaffolded feedback further accelerates learning of inquiry practices. These findings thus inform assessment designers and researchers that, if technology allows, adding real-time scaffolding can greatly benefit student learning of and performance on inquiry practices. This study contributes to research on the design of assessment systems in the following three ways. First, this study provides empirical evidence that assessment with automated, real-time scaffolding can effectively and efficiently improve students' learning. Second, this study further demonstrates how a science inquiry environment can foster student learning of practices even when scaffolding is not present. While students can learn within carefully designed environments without scaffolding, the rate at which they learn is slower relative to when scaffolding is provided. Lastly, the findings of this study inform designers and researchers of the benefits of adding real-time scaffolding to assessment systems in terms of the rate of student learning.

Acknowledgements. This research was supported by the Institute of Education Sciences (R305A120778) and National Science Foundation (1252477, 1629045) to Janice Gobert, Principal Investigator at Rutgers University.

References

1. Next Generation Science Standards Lead States: Next Generation Science Standards: for States, by States. National Academies Press, Washington (2013)
2. National Research Council: A Framework for K-12 Science Education: Practices, Crosscutting Concepts, and Core Ideas. National Academies Press, Washington (2012)
3. Gobert, J.D., Sao Pedro, M., Raziuddin, J., Baker, R.S.: From log files to assessment metrics: measuring students' science inquiry skills using educational data mining. J. Learn. Sci. **22**, 521–563 (2013)
4. Deters, K.M.: Student opinions regarding inquiry-based labs. J. Chem. Educ. **82**(8), 1178–1180 (2005)
5. Li, H., Gobert, J., Dicker, R.: Dusting off the messy middle: assessing students' inquiry skills through doing and writing. In: André, E., Baker, R., Hu, X., Rodrigo, M.M.T., du Boulay, B. (eds.) AIED 2017. LNCS (LNAI), vol. 10331, pp. 175–187. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-61425-0_15
6. Gobert, J.: US News & World Report, 13 May 2016. http://www.usnews.com/news/articles/2016-05-13/op-ed-educational-data-mining-can-enhance-science-education
7. Quellmalz, E.S., Timms, M.J., Silberglitt, M.D., Buckley, B.C.: Science assessments for all: integrating science simulations into balanced state science assessment systems. J. Res. Sci. Teach. **49**(3), 363–393 (2012)
8. van Joolingen, W.R., de Jong, T., Lazonder, A.W., Savelsbergh, E.R., Manlove, S.: Co-lab: research and development of an online learning environment for collaborative scientific discovery learning. Comput. Hum. Behav. **21**(4), 671–688 (2005)
9. Zapata-Rivera, D., Jackson, T., Liu, L., Bertling, M., Vezzu, M., Katz, Irvin R.: Assessing science inquiry skills using trialogues. In: Trausan-Matu, S., Boyer, K.E., Crosby, M., Panourgia, K. (eds.) ITS 2014. LNCS, vol. 8474, pp. 625–626. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-07221-0_84
10. Vygotsky, L.S.: Mind in Society: The Development of Higher Psychological Processes. Harvard University Press, Cambridge (1978)
11. Hmelo-Silver, C.E., Duncan, R.G., Chinn, C.A.: Scaffolding and achievement in problem-based and inquiry learning: a response to Kirschner, Sweller, and Clark (2006). Educ. Psychol. **42**(2), 99–107 (2007)
12. Quintana, C., Reiser, B.J., Davis, E.A., Krajcik, J., Fretz, E., Duncan, R.G., Kyza, E., Edelson, D., Soloway, E.: A scaffolding design framework for software to support science inquiry. J. Learn. Sci. **13**(3), 337–386 (2004)
13. Gobert, J., Moussavi, R., Li, H., Sao Pedro, M., Dickler, R.: Scaffolding students' on-line data interpretation during inquiry with Inq-ITS. In: Cyber-Physical Laboratories in Engineering and Science Education. Springer (in press)
14. Sao Pedro, M., Baker, R., Gobert, J.: Incorporating scaffolding and tutor context into Bayesian knowledge tracing to predict inquiry skill acquisition. In: Proceedings of the 6th International Conference on Educational Data Mining, pp. 185–192. EDM Society (2013)
15. Sao Pedro, M.: Real-time Assessment, Prediction, and Scaffolding of Middle School Students' Data Collection Skills Within Physical Science Simulations. Worcester Polytechnic Institute, Worcester (2013)

16. Moussavi, R., Gobert, J., Sao Pedro, M.: The effect of scaffolding on the immediate transfer of students' data interpretation skills within science topics. In: Proceedings of the 12th International Conference of the Learning Sciences, pp. 1002–1005. Scopus, Ipswich (2016)
17. Moussavi, R.: Design, Development, and Evaluation of Scaffolds for Data Interpretation Practices During Inquiry. Worcester Polytechnic Institute, Worcester (2018)
18. Li, H., Gobert, J., Dickler, R.: Automated assessment for scientific explanations in on-line science inquiry. In: Hu, X., Barnes, T., Hershkovitz, A., Paquette, L. (eds.) Proceedings of the 10th International Conference on Educational Data Mining, pp. 214–219. EDM Society, Wuhan (2017)
19. Gobert, J.D., Sao Pedro, M.A.: Digital assessment environments for scientific inquiry practices. In: Rupp, A.A., Leighton, J.P (eds.) The Wiley Handbook of Cognition and Assessment: Frameworks, Methodologies, and Applications, West Sussex, UK, pp. 508–534 (2017)

The Allocation of Time Matters to Students' Performance in Clinical Reasoning

Shan Li[1](✉), Juan Zheng[1], Eric Poitras[2], and Susanne Lajoie[1]

[1] Department of Educational and Counselling Psychology, McGill University,
845 Sherbrook Street West, Montreal, QC, Canada
{shan.li2,juan.zheng,susanne.lajoie}@mcgill.ca

[2] Department of Educational Psychology, University of Utah, Salt Lake City, UT 84112, USA
Eric.Poitras@utah.edu

Abstract. Understanding how students allocate their time to different learning behaviors, especially those that distinguish students' performances, can yield significant implications for the design of intelligent tutoring systems (ITS). Time on task is a typical indicator of students' self-regulated learning (SRL) and student engagement. In this paper, we analyze log file data to identify patterns in the behavior durations of 62 medical students in BioWorld, an ITS that supports them in regulating their diagnostic reasoning while solving complex patient cases. Results demonstrated that task complexity mediated the relationship between students' allocation of time and diagnostic performance outcomes. The high-performing students showed different patterns of time management with low-performing students when solving both simple and complex cases. Moreover, the durations of behaviors predicted students' performance in clinical reasoning.

Keywords: Self-regulated learning · Intelligent tutoring system
Clinical reasoning · Time allocation · Task complexity

1 Introduction

There is an increasing need for students in today's society to regulate certain aspects of their own learning, which has led to a substantial number of intervention studies intended for fostering self-regulated learning (SRL) [1]. One reason for developing students' SRL skills comes from the fact that lifelong learning is a necessity whereby individuals need to learn even after schooling [1]. A second reason is that students often fail to self-regulate their learning processes effectively and efficiently, especially in the contexts of solving naturalistic problems [2]. Research has demonstrated that students' self-regulatory processes are determinant factors of their achievement differences [3, 4]. For these reasons, intelligent tutoring systems (ITSs) have increasingly relied upon metacognitive tools designed to support SRL [5, 6, 7, 8]. The increased use of ITSs in turn provides researchers with a method for capturing students' dynamic SRL processes, as the systems can trace learners' activities as they engage in a task through log files [6, 7]. There is a growing consensus in the field that student performance is determined by the quantity and quality of their SRL activities [7, 9, 10]. However, little is known about the issue of efficiency (i.e. the extent to which time and effort is well used for an intended task) when

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 110–119, 2018.
https://doi.org/10.1007/978-3-319-91464-0_11

assessing students' self-regulatory processes and outcomes with technology-rich learning environments [2]. For this study, we investigated participants' SRL and medical diagnostic reasoning processes while solving patient cases in BioWorld [11]. Though there are different methods of examining SRL (e.g. questionnaires, verbal protocols, and eye-tracking), for the purpose of this study, we used log files to reveal how participants allocated their time and efforts. Specifically, the goal of this study was to examine if we could differentiate students' performance via their allocation of attentional resources during SRL processes as measured by the duration of learning behaviors.

2 Theoretical Framework

Self-regulated learning (SRL) is an active, constructive process whereby learners set learning goals and then attempt to control, monitor and regulate their cognitive and metacognitive processes in the service of these goals [12]. Students' SRL processes are crucial indicators of their performance in the context of ITSs [13]. In fact, analysis of trace log data to better understand students' learning processes has been an important area of educational research [13, 14]. Researchers agree on common characteristics of self-regulated learners [9]. For example, self-regulated learners can effectively identify relevant information, select appropriate problem-solving strategies, and adjust their actions based on the internal and external conditions. They also deliberately monitor their behaviors and self-reflect on their performances. However, the relationship between students' SRL processes and performance is neither obvious nor simple across different ITSs. For example, some researchers found that student performance was associated with the use of learning strategies (e.g. the deployment of surface and deep strategies) [9, 15], while some studies revealed that students' learning achievement could be predicted by the quantity of their SRL processes (e.g. the number of cognitive and metacognitive activities they applied) [4, 5]. High-performing students also differed from low-performing students in terms of behavioral patterns (e.g. sequencing of learning activities) [14, 16, 17].

While the aforementioned indicators, such as use of deep strategies, quantity of metacognitive activities, and expert-like problem-solving trajectories, all proved to be effective in predicting learners' performances, the extant literature on the relationship between students' performances and their allocation of time during SRL are quite mixed [2]. Studies from the field of expert-novice differences revealed that experts had faster access to domain-specific knowledge and strategies, thus they were more efficient than novices in solving a task [18]. Findings from [19] have also demonstrated that the time needed to solve a task was significantly reduced as students developed advanced self-regulated problem-solving skills. However, according to [20], the more time students worked on learning tasks during SRL processes with the hypermedia, the higher their performances. When delving into how learners allocate their time in concrete learning activities, the situation becomes more complex. A study conducted by [7] examined undergraduate student learning using MetaTutor, a hypermedia learning environment, and found that low performers spent most of their time on ineffective strategies. Though low performers did engage in certain key metacognitive processes they did not spend a great deal of time on them. High-performing students spent less time than low-performing students on identifying task-related information and setting their sub-goals [13], whereas a study by [21] suggested that high and low performers had no differences on the practice of self-regulated learning strategies.

In this paper, we are particular interested in how students allocate their time to different SRL behaviors while diagnosing virtual patient cases that differ in complexity, and whether the allocation of time distinguishes students' performances or not. Zimmerman's [22] model of SRL was applied as a theoretical framework in this study to examine students' self-regulatory processes, which consisted of three cyclical phases: forethought, performance and self-refection. We use this model because each of these phases relate to the clinical reasoning process that learners use while using BioWorld. In the forethought phase, learners analyze the task, set learning goals and determine which learning strategies to use based on the contextual environment requirements. The performance phase involves the actions taken to accomplish the task that are consciously monitored and controlled by students. Self-reflection refers to students' responses that involve systematically checking, judging and reflecting their performances.

Specifically, three research questions were formulated for this study:

(1) How do participants allocate their time in BioWorld to different clinical reasoning behaviors in terms of task complexity?
(2) Are there differences between high and low performers on the time that is allocated to different clinical reasoning behaviors?
(3) Do certain kinds of behavioral times predict participants' performances while diagnosing clinical cases?

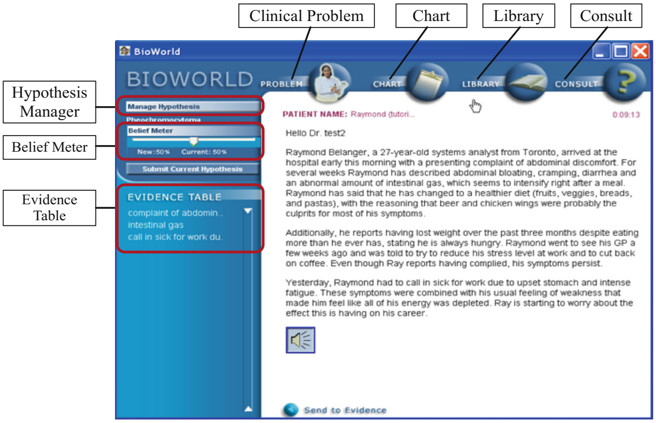
3 Methods

3.1 Participants

Sixty-two medical students from a large North American university volunteered to participate in this study. During the experiment, they were asked to diagnose two clinical cases. Of all 62 participants, 56 students completed the two cases while 6 students only accomplished one of the two cases. Specifically, 3 participants finished one case and the remaining 3 students completed the other one. Thus, to compare the case differences, 56 participants were used for analyses. However, 59 participants were used to find if high performers differed with low performers within a case.

3.2 Learning Context and Task

BioWorld is an intelligent tutor system designed to help medical students practice clinical reasoning skills in an authentic learning environment [11] (see Fig. 1). In BioWorld, each case begins with a description about a virtual patient and relevant symptoms. Based on the provided case information, students identify useful evidence by recalling their prior knowledge pertaining to the disease. Students then propose one or more hypotheses. To confirm or disconfirm their diagnoses, students can order lab tests to obtain further evidence or search an online library within BioWorld for additional explanations. After submitting a final diagnosis, students are asked to evaluate the relevance of the collected evidence with respect to their specific hypotheses, justify the probability of a hypothesis, and summarize their clinical reasoning processes in a case summary.



Hello Dr. test2

Raymond Belanger, a 27-year-old systems analyst from Toronto, arrived at the hospital early this morning with a presenting complaint of abdominal discomfort. For several weeks Raymond has described abdominal bloating, cramping, diarrhea and an abnormal amount of intestinal gas, which seems to intensify right after a meal. Raymond has said that he has changed to a healthier diet (fruits, veggies, breads, and pastas), with the reasoning that beer and chicken wings were probably the culprits for most of his symptoms.

Additionally, he reports having lost weight over the past three months despite eating more than he ever has, stating he is always hungry. Raymond went to see his GP a few weeks ago and was told to try to reduce his stress level at work and to cut back on coffee. Even though Ray reports having complied, his symptoms persist.

Yesterday, Raymond had to call in sick for work due to upset stomach and intense fatigue. These symptoms were combined with his usual feeling of weakness that made him feel like all of his energy was depleted. Ray is starting to worry about the effect this is having on his career.

Fig. 1. The main interface of the bioworld

In this study, participants were tasked with solving two patient cases in BioWorld, the Amy case and the Cynthia case. The correct diagnosis for each case was diabetes mellitus (type 1) and pheochromocytoma respectively. The Amy case was developed as an easy case while the Cynthia case was created as a difficult one.

3.3 Procedure

A training session was provided to teach participants how to use the BioWorld system before the experiment. They also had the chance to familiarize themselves with the system by solving a sample case in BioWorld. After the training session, participants were required to solve two clinical cases (i.e. the Amy case and the Cynthia case) independently for an appropriate total duration of 1.5 h. The order of patient case was randomized to counterbalance its effect on participants' performance.

3.4 Measures and Performance on Clinical Diagnosis

Eight diagnostic behaviors were coded according to the three self-regulated learning phases, forethought, performance and self-reflection. *Collecting evidence items (CO)* was coded as forethought. Three kinds of clinical reasoning behaviors were coded in the performance phase: *raising/managing hypotheses (RA), adding tests (AD)* and *searching the library* for information *(SE)*. The last phase (self-reflection) included *categorizing evidence/results (CA), linking evidences/results (LI), prioritizing evidences/results (PR)* and *summarization for final diagnosis (SU)*. The duration of each diagnostic behavior (i.e. how long a behavior lasts) for each participant was calculated based on the timestamps that recorded in the log files. To be specific, the duration of diagnostic behavior was obtained by subtracting the timestamp of the behavior itself from the subsequent one.

Participants' performances in solving the patient case were extracted from the BioWorld log files. Specifically, there were three types of performance metrics, namely, diagnostic confidence, accuracy, and efficacy. Based on the three performance indices, participants were grouped as the high performers and the low performers using the K-Nearest Neighbor classifier. To be specific, 36 and 23 participants were clustered as the low and the high performing students respectively when solving the Amy case. In terms of the Cynthia case, there were 30 low performing and 29 high performing participants.

3.5 Data Analysis

The collected data were analyzed using SPSS 23.0 and R [23]. Specifically, to address the third research question, the statistic modelling of conditional inference tree was applied. Conditional Inference Tree (CIT) is a tree-branched modelling method that developed to examine the relative importance of multiple potentially explanatory variables to a single response variable. It uses a machine-learning algorithm that is embedded in a conditional inference framework for recursive partitioning, which generates an inverted 'tree' of the variables in consequence [24]. CIT is not influenced by over-fitting and is applicable to different types of explanatory variables [25, 26]. It is also robust to multicollinearity, non-normality and non-linearity among explanatory variables [25].

4 Results

(1) How do participants allocate their time to different clinical reasoning behaviors in terms of task complexity?

Since each participant completed two different cases, paired t-tests were performed to identify if there were significant differences on the eight types of behavior durations between solving the Amy case and the Cynthia case. The results in Table 1 indicated that there were statistically significant differences on the total time of clinical diagnoses between the two cases, as well as on the time spent on *adding tests*. Specifically, participants took more time to solve the Cynthia case. And they applied more time on *adding tests* in solving the Cynthia case when compared with the Amy case.

(2) Are there differences between high and low performers on the time that is allocated to different clinical reasoning behaviors?

Table 1. All significant time differences between the case Amy and Cynthia

Time	Amy		Cynthia		*t*	*df*	Sig. (2-tailed)
	M	*SD*	*M*	*SD*			
Total time	1376.32	539.31	1560.39	583.87	−2.185	55	.033*
AD	306.43	219.68	451.43	316.11	−4.187	55	.000**

Note. *$p < .05$, **$p < .01$. AD = adding tests

To answer this question, a series of independent t-tests were performed (see Table 2). For the Amy case a statistically significant difference between high and low performers existed on the total time taken to make a clinical diagnosis. The high performers spent more time than the low performers. With regard to the concrete clinical reasoning behaviors, the high performers took more time on *categorizing evidences/results*, *prioritizing evidences/results* and *Summarization for final diagnosis* than low performers with statistically significant differences (see Table 2).

Table 2. All significant time differences between the high performers and low performers

Case	Time	Low		High		*t*	*df*	Sig. (2-tailed)
		M	*SD*	*M*	*SD*			
Amy	Total	1252.72	502.45	1584.91	531.59	2.42	57	.019*
	CA	67.78	32.15	111.57	42.77	4.48	57	.000**
	PR	51.58	32.31	96.52	52.46	3.69	57	.001**
	SU	107.69	94.22	275.43	190.51	3.93	57	.000**
Cynthia	RA	363.00	254.96	240.59	156.06	-2.22	57	.031*
	SE	85.53	115.28	37.72	58.94	-2.02	57	.050
	SU	129.43	92.84	190.10	130.24	2.07	57	.043*

Note. *$p < .05$, **$p < .01$. CA = Categorizing evidences/results, PR = Prioritizing evidences/results, SU = Summarization for final diagnosis, RA = raising/managing hypotheses, SE = searching the library.

There were significant differences between the high and low performers on the Cynthia case, on the time spent on *raising/managing hypotheses*, *searching the library* and *summarization for final diagnosis*. Specifically, the high performers spent less time on *raising/managing hypotheses* than the low performers. The high performing students also allocated less time *searching the library* than the low performing students, with a marginal statistical significance. With regard to *summarization for final diagnosis*, however, the high performers spent more time on it than the low performers.

(3) Do certain kind of behavioral time predict participants' performance while diagnosing clinical cases?

Statistical analysis was conducted in R to discover if there were any specific diagnostic behaviors that accounted for students' performance, taking advantage of the package 'party'. For the Amy case, the conditional inference tree model revealed that the clinical reasoning behavior of *summarization for final diagnosis* was the most influential factor in predicting participants' performance, followed by the behavior of *categorizing evidences/results* as shown on the left of Fig. 2. In terms of the Cynthia case the most influential variable to determine how to classify a high or low performer was the time spent on *raising/managing hypotheses* as shown on the right of Fig. 2. Furthermore, the diagnostic behaviors of *prioritizing evidences/results*, *summarization for final diagnosis* and *adding tests* also had the statistical power to differentiate the high and low performing groups.

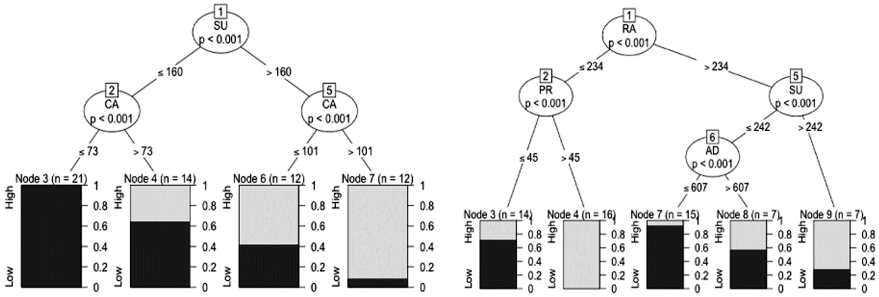


Fig. 2. Conditional inference tree assessing the relative importance of different clinical behaviors when solving the Amy case (left) and the Cynthia case (right) to participants' final performance (high performer vs. low performer). The splitting criterion was $P < .05$. p values are given below variable names. n indicates the number of observations. For the Amy case (left), Node 3 and Node 4 are primarily low performers [dark grey] (100% and 64.3% respectively), whereas node 7 is primarily high performers [light grey] (91.7%). Node 6 constitutes 58.3% high performers and 41.7% low performers; For the Cynthia case (right), Node 3 and node 7 are mainly low performers [dark grey] (71.4% and 93.3% respectively) while node 4 and Node 9 are mostly high performers (100% and 71.4% respectively). Node 8 consists of 57.1% low performers and 42.9% high performers.

5 Discussion

In general, this study found that case difficulty influenced the time spent solving cases for all performers. In other words, as the difficulty of medical case increased, participants spent relatively more time on clinical reasoning processes. Specifically, participants took significantly more time adding tests as the difficulty level increased. Besides, findings from this research suggested that when examining how students managed their SRL strategies during problem-solving processes, it's important to analyze how learners regulated their behaviors across the three phases of SRL, since different phases have different requirements for learners on the use of cognitive and metacognitive strategies. But it's also crucial to delve into specific behaviors within each phase of SRL, otherwise some details would be ignored from the analyses.

With respect to the simpler case of Amy, the high performers spent significantly more time than the low performers. This was in line with the findings from [20] which argued that the more time students regulated their own problem-solving, the better the outcomes. However, there were no statistically significant differences between the high- and low-performing learners in terms of the time spent solving the Cynthia case (i.e. the difficult case), indicating that task complexity mediated the relationship between problem-solving time and student performance. As the difficulty level of patient case increased, total time increased but did not differentiate between high and low performance. Rather, performance differences could be attributed to where students spent the most time, i.e., different allocation of time to different SRL behaviors. When examining the time differences on specific diagnostic behaviors, this study found that the high performing groups took more time on *categorizing evidences/results*, *prioritizing* *evidences/results* and *summarization for final diagnosis* than low performing groups to solve the Amy case, revealing that high performers allocated more time on deep learning strategies. It's important to notice that these three kinds of behaviors all belong to the self-reflection phase of SRL, since they essentially involved the cognitive and metacognitive activities of checking, judging and reflecting upon their learning outcomes. Moreover, the conditional inference tree analysis also revealed that *summarization for final diagnosis* and *categorizing evidences/results* were the two most important behaviors to predict participants' final performances when solving the Amy case. This was in accordance with the observations of [7], claiming that low performers spent shorter periods of time on key metacognitive processes.

In terms of the more complex case of Cynthia, there were significant differences in the time spent on specific behaviors that occurred during the performance phase and the self-reflection phase of SRL between the high and low performers. The low performers took more time in the performance phase, while the high performers spent more time in the phase of self-reflection. Specifically, the high performers spent less time on *raising/managing hypotheses*, *searching the library*. But the high performers spent significantly more time on *summarization for final diagnosis* than the low performers, which was consistent with results associated with the simple case. These results have corroborated previous findings that low performers spent most of their time on ineffective strategies, while high performers selectively focused on effective learning activities [6, 7, 13]. Furthermore, the diagnostic behaviors of *raising/managing hypotheses, summarization for final diagnosis,* aligned with *prioritizing evidences/results*, and *adding tests*, played a crucial rule on participants' performance. The amount of time spent in specific behaviors that occurred during the performance and reflection phase were different from those of students solving the Amy case. The reason perhaps lies in the fact that high performers needed to adjust their actions and monitoring activities more efficiently as the task difficulty increased, and therefore adapted their use of different strategies to better achieve their goals [2].

6 Conclusion

This study examined how students allocated their time to different SRL behaviors while diagnosing virtual patient cases that differ in complexity, and whether the allocation of time distinguished between low and high students' performances. The findings from this research demonstrated that task complexity was an important mediator in the relationship between students' allocation of time and their performances, which provided new evidence to inform the mixed results evident in the extant literature [2]. Moreover, the allocation of time could also predict students' performances.

This research provided important implications for the development of ITSs designed as metacognitive tools. Task complexity is an essential consideration when developing SRL tools within an ITS, since different levels of tasks require different learning strategies. Furthermore, our findings suggest that adaptive scaffolds for promoting SRL processing should consider the efficient use of different strategies that are critical to successful task completion.

References

1. van Ewijk, C.D.: Assessing students' acquisition of self-regulated learning skills using meta-analysis. In: Azevedo, R., Aleven, V. (eds.) Handbook of Self-Regulation of Learning and Performance, pp. 376–390. Routledge, Abingdon (2011)
2. Taub, M., Azevedo, R., Bradbury, A.E., Millar, G.C., Lester, J.: Using sequence mining to reveal the efficiency in scientific reasoning during STEM learning with a game-based learning environment. Learn. Instr. **54**, 93–103 (2017)
3. Zimmerman, B.J., Schunk, D.H.: Self-regulated learning and performance: an introduction and an overview. In: Handbook of Self-Regulation of Learning and Performance, pp. 1–12 (2011)
4. Poitras, E.G., Lajoie, S.P.: A domain-specific account of self-regulated learning: the cognitive and metacognitive activities involved in learning through historical inquiry. Metacogn. Learn. **8**, 213–234 (2013). https://doi.org/10.1007/s11409-013-9104-9
5. Biswas, G., Segedy, J.R., Bunchongchit, K.: From design to implementation to practice a learning by teaching system: Betty's brain. Int. J. Artif. Intell. Educ. **26**, 350–364 (2016). https://doi.org/10.1007/s40593-015-0057-9
6. Lajoie, S.P., Naismith, L., Poitras, E., Hong, Y.-J., Cruz-Panesso, I., Ranellucci, J., Mamane, S., Wiseman, J.: Technology-rich tools to support self-regulated learning and performance in medicine. In: Azevedo, R., Aleven, V. (eds.) International Handbook of Metacognition and Learning Technologies. SIHE, vol. 28, pp. 229–242. Springer, New York (2013). https://doi.org/10.1007/978-1-4419-5546-3_16
7. Azevedo, R., Johnson, A., Chauncey, A., Burkett, C.: Self-regulated learning with metatutor: advancing the science of learning with metacognitive tools. In: Khine, M., Saleh, I. (eds.) New Science of Learning: Cognition, Computers and Collaboration in Education, pp. 225–247. Springer, New York (2010). https://doi.org/10.1007/978-1-4419-5716-0_11
8. Li, S., Zheng, J.: The effect of academic motivation on students' English learning achievement in the eSchoolbag-based learning environment. Smart Learn. Environ. **4**, 3 (2017). https://doi.org/10.1186/s40561-017-0042-x
9. Zimmerman, B.J.: Becoming a self-regulated learner: an overview. Theory Pract. **41**, 64–70 (2002). https://doi.org/10.1207/s15430421tip4102_2
10. Roll, I., Winne, P.H.: Understanding, evaluating, and supporting self-regulated learning using learning analytics. J. Learn. Anal. **2**, 7–12 (2015). https://doi.org/10.18608/jla.2015.21.2
11. Lajoie, S.P.: Developing professional expertise with a cognitive apprenticeship model: examples from avionics and medicine. In: Ericsson, K.A. (ed.) Development of Professional Expertise: Toward Measurement of Expert Performance and Design of Optimal Learning Environments, pp. 61–83. Cambridge University Press, New York (2009)
12. Pintrich, P.R.: The role of goal orientation in self-regulated learning. Handb. Self-Regul. 451–502 (2000). https://doi.org/10.1016/b978-012109890-2/50043-3
13. Bouchet, F., Azevedo, R., Kinnebrew, J.S., Biswas, G.: Identifying students' characteristic learning behaviors in an intelligent tutoring system fostering self-regulated learning. In: Proceedings of the 5th International Conference on Education Data Mining (EDM 2012), pp. 65–72 (2012)
14. Hadwin, A.F., Nesbit, J.C., Jamieson-Noel, D., Code, J., Winne, P.H.: Examining trace data to explore self-regulated learning. Metacogn. Learn. **2**, 107–124 (2007). https://doi.org/10.1007/s11409-007-9016-7
15. Kostons, D., van Gog, T., Paas, F.: Training self-assessment and task-selection skills: a cognitive approach to improving self-regulated learning. Learn. Instr. **22**, 121–132 (2012). https://doi.org/10.1016/j.learninstruc.2011.08.004

16. Bannert, M., Reimann, P., Sonnenberg, C.: Process mining techniques for analysing patterns and strategies in students' self-regulated learning. Metacogn. Learn. **9**, 161–185 (2014). https://doi.org/10.1007/s11409-013-9107-6
17. Poitras, E.G., Doleck, T., Lajoie, S.P.: Towards detection of learner misconceptions in a medical learning environment: a subgroup discovery approach. Educ. Technol. Res. Dev. **66**, 129–145 (2017)
18. Ericsson, K.A., Charness, N.: Expert performance: its structure and acquisition. Am. Psychol. **49**, 725–747 (1994). https://doi.org/10.1037/0003-066X.49.8.725
19. Lazakidou, G., Retalis, S.: Using computer supported collaborative learning strategies for helping students acquire self-regulated problem-solving skills in mathematics. Comput. Educ. **54**, 3–13 (2010). https://doi.org/10.1016/j.compedu.2009.02.020
20. Narciss, S., Proske, A., Koerndle, H.: Promoting self-regulated learning in web-based learning environments. Comput. Hum. Behav. **23**, 1126–1144 (2007). https://doi.org/10.1016/j.chb.2006.10.006
21. Vighnarajah, Wong, S.L., Abu Bakar, K.: Qualitative findings of students' perception on practice of self-regulated strategies in online community discussion. Comput. Educ. **53**, 94–103 (2009). https://doi.org/10.1016/j.compedu.2008.12.021
22. Zimmerman, B.J.: Self-regulated learning and academic achievement: an overview. Educ. Psychol. **25**, 3–17 (2010). https://doi.org/10.1207/s15326985ep2501
23. R Core Team: R: A Language and Environment for Statistical Computing (2014)
24. Hothorn, T., Hornik, K., Zeileis, A.: Unbiased recursive partitioning: a conditional inference framework. J. Comput. Graph. Stat. **15**, 651–674 (2006). https://doi.org/10.1198/106186006X133933
25. Hothorn, T., Hornik, K., Strobl, C., Zeileis, A.: Party: a laboratory for recursive partytioning. R Package version 0.9-0. 37 (2006). 10.1.1.151.2872
26. Strobl, C., Boulesteix, A.L., Zeileis, A., Hothorn, T.: Bias in random forest variable importance measures: illustrations, sources and a solution. BMC Bioinform. **8**, 25 (2007). https://doi.org/10.1186/1471-2105-8-25

Empirical Investigation of Cognitive Load Theory in Problem Solving Domain

Kazuhisa Miwa[1](✉), Hitoshi Terai[2], and Kazuaki Kojima[3]

[1] Nagoya University, Nagoya 464-8601, Japan
miwa@is.nagoya-u.ac.jp
[2] Kindai University, Iizuka 820-8555, Japan
[3] Teikyo University, Utsunomiya 320-8551, Japan

Abstract. The cognitive load theory has been mainly investigated in declarative knowledge learning, typically learning with hyper-media material. In this study, the preceding findings are examined in problem solving domain with a different type of experimental task such as Reversi game. The experimental results were consistent with preceding studies, showing that extraneous cognitive load is harmful to the learning process, but the effects of intrinsic load are subject to debate. Additionally, the participants correctly evaluated each cognitive load, using a questionnaire. In addition, it was confirmed that the subjective evaluation predicted learning outcomes.

Keywords: Cognitive load theory · Intrinsic load · Extraneous load
Germane load

1 Introduction

The cognitive load theory (CLT) plays a central role in designing learning environments [20,21]. It distinguishes three types of cognitive loads: intrinsic, extraneous, and germane. The theory has been mainly examined in the domain of declarative learning. In typical situations for examination, participants engaged in learning activities with hyper-media learning materials in which the burdened cognitive loads were manipulated. The findings indicate that extraneous load is harmful [5,12,13,21], but germane load is helpful for learning [1,16,17,21]. In addition, intrinsic load should be controlled on the basis of the tradeoff between participants' expertise and the difficulty of the learning tasks [18].

In the present study, we examined CLT in a different learning domain. We used an 8-by-8 Reversi board game as the learning task. The skills for performing such a board game are relatively different from those in the learning domain in which the theory has been examined so far. One crucial skill for the game is the capability to search for the problem space necessary to execute the best move. As players master the game, their ability to search and predict future states of problem solving develop further. These types of procedural skills are

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 120–129, 2018.
https://doi.org/10.1007/978-3-319-91464-0_12

commonly required in various types of problem solving, and relate to general problem solving heuristics such as the mean-ends analysis [19].

Another type of development occurs in the acquisition of perceptual skills. Many cognitive science studies on problem solving have indicated that game experts acquire various types of perceptual schema to perform rapid and accurate recognition of the state of the game as it is played [3]. Such perceptual skills are established on the basis of perceptual schema as crucial knowledge constructed through the processes of acquiring expertise.

In the present study, we define intrinsic, extraneous, and germane cognitive loads when performing the Reversi game, and require participants to perform an experiment in which two primary loads, intrinsic and extraneous, which have been focused since the beginning of CLT studies, are manipulated [20]. Generally, the intrinsic load is defined as the basic cognitive load required to perform a particular task. As mentioned, the primary cognitive activities for performing the current task involve searching for the problem space. Therefore, in the Reversi experiment, we define the intrinsic load as the load for finding the next best move in each game state. To find the move, participants are required to search the successive problem states. The extraneous load is defined as the cognitive load unrelated to and hence wasted in primary cognitive activities. Thus, we define extraneous load as the load that disturbs search activities. To manipulate extraneous load in the experiment, we use different game discs rather than black and white discs that increase the cognitive load to understand each game status.

Finally, the germane load is not manipulated in the experiment, and is treated as the different cognitive load that emerges from the intrinsic load, but provides positive effects for learning [1,16,17,21]. In the present study, we define the germane load as necessary to discover the heuristic knowledge required to win the game. Many types of such knowledge are recognized such as “occupying the four corners as rapidly as possible.” To find such heuristics, participants engage in meta cognitive activities that reflect their game playing, causing additional cognitive loads while performing the task.

The first objective of this study is to understand the relation between such cognitive loads and learning effects. We capture the learning effects on the basis of the increases in test scores from pre- to post-test.

The second objective is drawn in the context of the development of measurement methodology in CLT studies. It is common to use a questionnaire to elicit the subjective evaluations of participants; questionnaires typically comprise questions related to one of the three types of cognitive loads. However, the reliability of this subjective evaluation method has been called into question as a result of multiple evaluations using questionnaires exhibiting inconsistencies [6]. Some studies confirmed the validity of this methodology, but others did not.

In this study, we designed a questionnaire for measurement on the basis of the definition of intrinsic, extraneous, and germane loads as described above, and examined whether each cognitive load could be measured on the basis of the subjective responses to the questionnaire.

Additionally, we examined such subjective estimation as a predictor of learning effects, with particular attention to the consistency between the subjective estimations and the learning effects of the two experimental factors.

2 Learning Task

2.1 Task

The task used in this study involved an 8-by-8, computer-based Reversi game, for which a Reversi-based learning environment was developed by the authors [14,15]. Participants played 8-by-8 Reversi games on a computer against a computerized opponent (i.e., opponent agent) in the experimental environment. Participants were assisted by a partner, also computerized (i.e., partner agent), to selecting winning moves. The opponent agent and the partner agent were both controlled by Edax, a Reversi engine, which suggested the best moves by assessing future states of the game. The partner agent typically recommends candidate moves among valid squares before the participant makes a move.

2.2 Questionnaire for Cognitive Load Evaluation

We developed a new questionnaire for cognitive load evaluation referring to the questionnaires used in previous studies [5,11]. The questionnaire used in the present study consists of ten items. Example question items for intrinsic, extraneous, and germane loads are: it is difficult to search for the best move, it is difficult to understand the arrangements of the discs on the board, and I make great effort to find heuristics for wining, respectively.

3 Experiment

3.1 Procedure

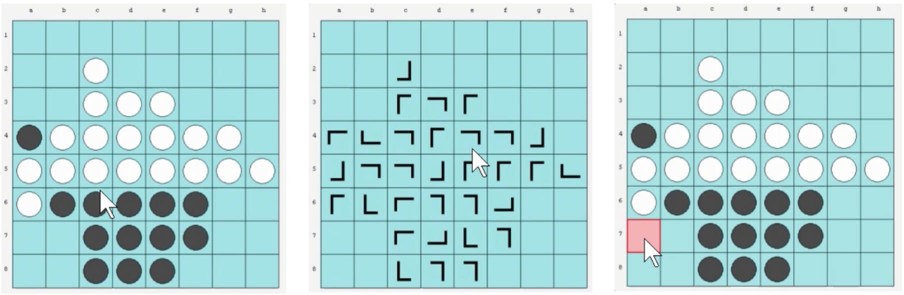
In order to determine the baseline for the measurement of learning gains, the participants were involved in a pre-test, which consisted of 12 problems. Following this, the participants took part in the learning (training) phase, which involved 16 games for training. This phase was set up such that the participants had access only to games that were already in progress; as a result, nearly half of the discs were already placed on the board. From the middle toward final stage, the participants played each game against the opponent agent, in some of experimental conditions, while receiving the guidance information (i.e., the best move) from the partner agent. At the final stage, they received the result, victory or defeat, with the number of discs they had gotten.

The learning phase consisted of four blocks, and the participants were required to play four games in each block. A set of winning strategies is proposed; and the training for each block enabled the participants to learn one of the strategies. The discs were arranged in an identical manner for the first three games in each block, whereas the arrangement was altered for the fourth (final) game. The participants were then required to work with the questionnaire designed to evaluate cognitive loads; they were also required to take part in the post-test, which consisted of the same 12 problems as the pre-test.

3.2 Manipulated Factors

The following two factors were applied for manipulation: (i) the disc representation factor and (ii) the guidance information factor. The first factor was expected to manipulate the extraneous load, whereas the second factor was expected to manipulate the intrinsic load, respectively.

Disc Representation Factor. Figure 1 presents a sample disc arrangement typical of the Black and White, and the L and rL (reversal L) conditions.



(a) Black and White condition (b) L and reversal L condition (c) Guidance presentation condition

Fig. 1. Screenshots of the game board in the Black and White, L and rL (reversal L), and guidance presentation conditions.

When the Black and White condition was considered, the Black and White discs were used in the arrangement, whereas when the L and rL condition was considered, the Ls or rotated Ls (black discs) and the mirror reversal Ls or rotated reversal Ls (white discs) were used in the arrangement. In order to perceive the status of the disc arrangement and decide the best move in the L and rL condition, participants had to imagine the rotation of the L or reversal L images during each trial, thus causing a significant extraneous load. As a result, the L and rL condition increased the extraneous load more than the Black and White condition.

Guidance Information Factor. For each trial of the game, the main task was to choose the best winning move. In order to do so, the participants had to understand the status of the disc arrangement, search the problem space, and estimate the best move, thus increasing intrinsic load. The computerized partner agent suggested the best move to the participants in the guidance presentation condition (see Fig. 1), whereas under the no guidance condition, no such information was presented. This suggests that the intrinsic load of the participants was lower in the guidance presentation condition than in the no guidance condition.

3.3 Learning Gains

Pre- and post-tests were conducted to evaluate the learning gains, and each test consisted of the same 12 problems. In each problem, the participants were presented with a disc arrangement, after which they were required to determine the best possible move. The 12 problems were grouped into the following three categories, each of which consisted of four problems. In identical problems, this disc arrangements presented here were identical to those used in the training phase. In near transfer problems, the disc arrangements used for the learning phase were modified. More specifically, they were rotated 90, 180, or 270 degrees from their original position or mirror-reversed from the rotated arrangements. Finally, in far transfer problems, new disc arrangements were presented for this category. The participants were able to determine the best possible move based on the strategies they were trained in during the learning phase.

As the number of problems in each category was four, the full score was also determined to be four. In the present study, the difference between the pre-test and post-test scores, more particularly, the increase in the post-test scores, were used as learning gains.

3.4 Participants

81 undergraduates from Nagoya University participated in this study. Although all the participants had played Reversi prior to their involvement in the study, they were not experts. The participants were divided into four groups: 21, 19, 20, and 21 participants were assigned to each of no guidance and Black/White, no guidance and L/rL, guidance presentation and Black/White, and guidance presentation and L/rL conditions, respectively.

4 Results

4.1 Manipulation Check

We assume that the disc representation factor manipulates the extraneous load, and the guidance information factor manipulates the intrinsic load, respectively. To confirm this premise, we analyzed the following two indexes.

Disc Representation Factor. The preceding studies have confirmed that high extraneous load makes many errors in task performance. Therefore, we analyzed the number of errors (invalid) moves in which the participants received alarms from the system. Table 1 shows the average number of errors in each game.

Table 1. Number of error moves in the learning phase

No guide Black/White	No guide L/rL	Guide Black/White	Guide L/rL
1.82 (0.29)	13.47 (0.81)	0.35 (0.08)	3.14 (0.83)

Table 1 shows that errors were extremely large in the no guidance and L/rL condition in which the enormous extraneous load expects to emerge. A two (guidance information: guide and no guide) × two (disc representation: Black/white and L/rL) ANOVA revealed significant interaction between the two factors ($F(1, 77) = 50.5$, $p < 0.01$). The simple main effects of the disc representation factor in both guide and no guide conditions revealed significance ($F(1, 77) = 174.6$, $p < 0.01$; $F(1, 77) = 10.0$, $p < 0.01$). This result supports that the L and rL disc representation successfully caused larger extraneous load.

Guidance Information Factor. As mentioned above, in the current learning task, the primary cognitive activity causing the intrinsic load to perform the task was to search for the problem space. This means that a larger intrinsic load expects participants to take greater time to decide the next move, which results in greater response time to reach a decision. Table 2 shows the average response time required by the participants to decide each move.

Table 2. Response time (msec) in each decision for next move

No guide Black/White	No guide L/rL	Guide Black/White	Guide L/rL
5958.5 (481.3)	7265.0 (519.3)	3987.4 (388.0)	4544.1 (418.8)

The same ANOVA revealed a significant main effect of the guidance information factor ($F(1, 77) = 25.1$, $p < 0.01$), but neither the main effect of the disc representation factor nor the interaction of the two factors reached a significant level ($F(1, 77) = 4.0$, n.s.; $F(1, 77) < 1$, n.s.). This result supports that the guidance information factor successfully manipulated the intrinsic cognitive load in the experiment.

4.2 Learning Effects

Table 3 shows the increase in the test scores from pre- to post-test in the identical, near transfer, and far transfer problem categories.

In each of the problem categories, the same 2 × 2 ANOVA was performed. In the identical problem category, it detects a significant main effect of the disc representation factor ($F(1, 77) = 10.3$, $p < 0.01$), but neither the main effect of the guidance information factor nor the interaction between the two factors reached a significant level ($F(1, 77) < 1$, n.s.; $F(1, 77) = 2.5$, n.s.).

Table 3. Increase in test scores from pre- to post-test

	No guide Black/White	No guide L/rL	Guide Black/White	Guide L/rL
Identical	0.90 (0.28)	0.50 (0.28)	1.16 (0.24)	−0.05 (0.16)
Near transfer	0.76 (0.24)	0.60 (0.25)	1.21 (0.24)	0.29 (0.22)
Far transfer	0.33 (0.35)	0.75 (0.32)	0.58 (0.27)	0.24 (0.29)

In the near transfer problems, we obtained a similar result, indicating that there was a significant main effect of the disc representation factor ($F(1, 77) = 4.9$, $p < 0.05$), but neither the main effect of the guidance information factor nor the interaction between the two factors reached a significant level ($F(1, 77) < 1$, n.s.; $F(1, 77) = 2.4$, n.s.).

In the far transfer problems, none of the main effect of the disc representation factor, the main effect of the information guidance factor, or the interaction of the two factors was detected ($F(1, 77) < 1$, n.s.; $F(1, 77) < 1$, n.s.; $F(1, 77) = 1.4$, n.s.).

4.3 Subjective Estimation

Next, we examined whether the participants evaluate cognitive loads correctly when the extraneous and intrinsic cognitive loads are manipulated. Table 4 presents the results of the questionnaire used to measure each type of cognitive load.

Table 4. Results of participants' subjective evaluation for the three cognitive loads

	No guide Black/White	No guide L/rL	Guide Black/White	Guide L/rL
Intrinsic	4.11 (0.17)	4.56 (0.12)	3.70 (0.19)	3.75 (0.26)
Extraneous	2.19 (0.15)	4.37 (0.08)	1.58 (0.10)	4.54 (0.09)
Germane	4.19 (0.09)	3.51 (0.17)	4.11 (0.11)	3.08 (0.18)

In the intrinsic load, a two (guidance information: Guide and No guide) × two (disc representation: Black/white and L/rL) ANOVA revealed a significant main effect of the guidance information factor ($F(1, 77) = 9.59$, $p < 0.01$), but neither the main effect of the disc representation factor nor the interaction of the two factors reached a significant level ($F(1, 77) = 1.58$, n.s.; $F(1, 77) = 1.07$, n.s.).

In the extraneous load, the same ANOVA revealed a great significant main effect of the disc representation factor ($F(1, 77) = 520.43$, $p < 0.01$) but the main effect of the guidance information factor did not reach a significant level ($F(1, 77) = 3.79$, n.s.). The interaction of the two factors, however, was found to be significant ($F(1, 77) = 12.13$, $p < 0.01$).

In the germane load, the same ANOVA revealed a significant main effect of the disc representation factor ($F(1, 77) = 35.13,\ p < 0.01$), but neither the main effect of the guidance information factor nor the interaction of the two factors reached a significant level ($F(1, 77) = 3.21$, n.s.; $F(1, 77) = 1.43$, n.s.).

4.4 Subjective Estimation and Learning Gains

In Tables 3 and 4, we obtained two results referring to two dependent valuables, learning effects and subjective evaluation scores. Next, we sought to identify the type of cognitive load that contributes to learning effects, specifically, we analyzed correlations between the two indexes. We analyzed the relations between the evaluation scores of each cognitive load in the questionnaire and the increases in the post-test scores as compared to the pre-test scores. Table 5 presents the results of this analysis. We found a positive correlation between the germane load and the learning gains, but only in the identical problems that were tested. Negative correlations were found between the extraneous load and the learning gains in the identical and near transfer problems. However, the relations were not noticeable when the far transfer problems were tested.

Table 5. Correlation of the evaluation scores of each cognitive load from the questionnaire and the difference between the pre-test and post-test scores

	Intrinsic	Extraneous	Germane
Identical	n.s.	$r = -0.385,\ p < 0.01$	$r = 0.279,\ p < 0.05$
Near transfer	n.s.	$r = -0.294,\ p < 0.01$	n.s.
Far transfer	n.s.	n.s.	n.s.

5 Discussion and Conclusions

First, we consider the first research question posed in introduction. In the Reversi experiment, we confirmed successful manipulation for intrinsic and extraneous cognitive loads. As indicated in Sect. 4.2, increasing of the extraneous load lowered learning gains, whereas the amount of intrinsic load showed no effects of the learning outcomes. This finding is consistent in the subjective evaluation in Sect. 4.4, which means that subjective evaluation functions as a predictor for learning outcomes.

In terms of the second research question, the results in Sect. 4.3 confirmed that the participants correctly evaluated both intrinsic and extraneous loads. Further analysis showed that there was no correlation between the subjective evaluations of the intrinsic and extraneous loads ($F(1, 79) = 2.36$, n.s.), meaning that they estimated each cognitive load independently. On the contrary, we found a substantial negative correlation between the extraneous and germane loads ($r = -0.481,\ F(1, 79) = 23.88,\ p < 0.01$). This analysis implies that our experimental manipulation of disc representation caused a heavy extraneous load, and limited working memory resources to which the germane load is assigned for learning.

The negative cost of the extraneous load mentioned above is consistent with results obtained from preceding studies on CLT [5,12,13,21]. On the other hand, the relation between the intrinsic load and learning gains is still unclear [2,6,9,18]. CLT has two perspectives on the function of intrinsic load: learning activities emerge together with, or independently from, the intrinsic load.

From the part-of-intrinsic view, typical procedural learning occurs with an increase in the capabilities for looking ahead to a successive game status while searching the problem space for the best possible move to win. On the basis of the definition of intrinsic load in this study, these types of learning activities emerge with the intrinsic load. Conversely, from the independent-from-intrinsic view, crucial learning could be developed by finding accurate heuristics for winning games. This type of knowledge is expected to be found through reflective thinking. When participants were in the guidance presentation condition, they considered why the presented move was the best one. Such thinking is characterized as a meta cognitive activity. The guidance presentation condition compressed the intrinsic load and widened the cognitive resources to which this type of germane cognitive load was assigned. In the current experiment, the increase of intrinsic load might activate the former type of learning activities whereas the decrease of the intrinsic load might activate the latter learning. Such two aspects of the functions of intrinsic load may negate the relation between the intrinsic load and learning gains.

Finally, although substantial increases were found in the test scores from pre- to post-test in Black and White conditions, the increases were only found in the identical and near transfer problem categories. Meanwhile, no increases were found in the far transfer problems. Overall, skill acquisition of playing games needs much training and the expertise as a game player needs to take long period [8]. In our experiment, only sixteen games were performed for training. There is a possibility that this limitation did not cause learning effects in the far transfer problems.

We confirmed the validity of CLT using the Reversi game as an experimental task. The findings of this study were almost consistent with those of previous studies. CLT has provided design principles: the design of hyper media [10] such as e-learning systems, websites [4], and online newspapers [7]. Our study indicated a possibility that such principles could be applied to other expanded learning domains.

Acknowledgement. This work was supported by JSPS KAKENHI Grant Numbers 15H02927 and 16H02353.

References

1. Ayres, P., van Gog, T.: State of the art research into cognitive load theory. Comput. Hum. Behav. **25**, 253–257 (2009)

2. Beckmann, J.F.: Taming a beast of burden-on some issues with the conceptualisation and operationalisation of cognitive load. Learn. Instr. **20**(3), 250–264 (2010)
3. Chase, W.G., Simon, H.A.: Perception in chess. Cogn. Psychol. **4**(1), 55–81 (1973)
4. Chevalier, A., Kicka, M.: Web designers and web users: influence of the ergonomic quality of the web site on the information search. Int. J. Hum. Comput. Stud. **64**(10), 1031–1048 (2006)
5. Cierniak, G., Scheiter, K., Gerjets, P.: Explaining the split-attention effect: is the reduction of extraneous cognitive load accompanied by an increase in germane cognitive load? Comput. Hum. Behav. **25**(2), 315–324 (2009)
6. De Jong, T.: Cognitive load theory, educational research, and instructional design: some food for thought. Instr. Sci. **38**(2), 105–134 (2010)
7. Debue, N., van de Leemput, C.: What does germane load mean? An empirical contribution to the cognitive load theory. Front. Psychol. **5**, 1099 (2014)
8. Ericsson, K.A.: The Road to Excellence: The Acquisition of Expert Performance in the Arts and Sciences, Sports, and Games. Psychology Press, Philadelphia (2014)
9. Haji, F.A., Rojas, D., Childs, R., Ribaupierre, S., Dubrowski, A.: Measuring cognitive load: performance, mental effort and simulation task complexity. Med. Educ. **49**(8), 815–827 (2015)
10. Hollender, N., Hofmann, C., Deneke, M., Schmitz, B.: Integrating cognitive load theory and concepts of human–computer interaction. Comput. Hum. Behav. **26**(6), 1278–1288 (2010)
11. Leppink, J., Paas, F., van Gog, T., van der Vleuten, C.P., van Merrinboer, J.J.: Effects of pairs of problems and examples on task performance and different types of cognitive load. Learn. Instr. **30**, 32–42 (2014)
12. Mayer, R.E., Moreno, R.: A split-attention effect in multimedia learning: evidence for dual processing systems in working memory. J. Educ. Psychol. **90**(2), 312–320 (1998)
13. Mayer, R.E., Moreno, R.: Nine ways to reduce cognitive load in multimedia learning. Educ. Psychol. **38**(1), 43–52 (2003)
14. Miwa, K., Kojima, K., Terai, H.: An experimental investigation on learning activities inhibition hypothesis in cognitive disuse atrophy. In: Proceedings of 7th International Conference on Advanced Cognitive Technologies and Applications (Cognitive 2015), pp. 66–71 (2015)
15. Miwa, K., Kojima, K., Terai, H.: Measuring cognitive loads based on the mental chronometry paradigm. In: Proceedings of 8th International Conference on Advanced Cognitive Technologies and Applications (Cognitive 2016), pp. 38–41 (2016)
16. Paas, F.G., Van Merrienboer, J.J.: Variability of worked examples and transfer of geometrical problem-solving skills: a cognitive-load approach. J. Educ. Psychol. **86**(1), 122–133 (1994)
17. Paas, F., van Gog, T.: Optimising worked example instruction: different ways to increase germane cognitive load. Learn. Instr. **16**(2), 87–91 (2006)
18. Schnotz, W., Kürschner, C.: A reconsideration of cognitive load theory. Educ. Psychol. Rev. **19**(4), 469–508 (2007)
19. Simon, H.A.: Human Problem Solving. Prentice Hall, Upper Saddle River (1972)
20. Sweller, J.: Cognitive load during problem solving: effects on learning. Cogn. Sci. **12**(2), 257–285 (1988)
21. Sweller, J., Van Merrienboer, J.J., Paas, F.G.: Cognitive architecture and instructional design. Educ. Psychol. Rev. **10**(3), 251–296 (1998)

Data-Driven Learner Profiling Based on Clustering Student Behaviors: Learning Consistency, Pace and Effort

Shirin Mojarad[1](✉), Alfred Essa[1], Shahin Mojarad[1], and Ryan S. Baker[2]

[1] McGraw-Hill Education, New York, USA
{shirin.mojarad,Alfred.essa, s.a.mojarad}@mheducation.com
[2] University of Pennsylvania, Philadelphia, USA
Rybaker@upenn.com

Abstract. While it is important to individualize instruction, identifying and implementing the right intervention for individual students is too time-consuming for instructors to do manually in large classes. One approach to addressing this challenge is to identify groups of students who would benefit from the same intervention. As such, this work attempts to identify groups of students with similar academic and behavior characteristics who can benefit from the same intervention. In this paper, we study a group of 700 students who have been using ALEKS, a Web-based, adaptive assessment and learning system. We group these students into a set of clusters using six key characteristics, using their data from the first half of the semester, including their prior knowledge, number of assessments, average days and score increase between assessments, and how long after the start of the class the student begins to use ALEKS. We used mean-shift clustering to select a number of clusters, and k-mean clustering to identify distinct student profiles. Using this approach, we identified five distinct profiles within these students. We then analyze whether these profiles differ in terms of students' eventual degree of content mastery. These profiles have the potential to enable institutions and instructors using ALEKS to identify students in need and devise and implement appropriate interventions for groups of students with similar characteristics and needs.

Keywords: Group intervention · ALEKS · Clustering · Student profiling Grit

1 Introduction

Interventions delivered by instructors can change students' course of learning, guiding them to improve their outcomes [1]. However, despite the success of specific broad-based interventions [2], it is unlikely that any single intervention will be ideal for all students. A major limitation to the development of broad-based, classroom-wide interventions is that students' characteristics are highly variable from student to student. This variability has made it difficult to identify and apply the right intervention in classroom and online platforms supporting instructors.

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 130–139, 2018.
https://doi.org/10.1007/978-3-319-91464-0_13

This realization has led educators to consider personalized interventions, where each individual receives an intervention tailored to their needs. However, it is time-consuming for instructors to identify and implement the right intervention for individual students, especially if they have to do so manually in large classes. Fortunately, students are not fully unique either; what enhances learning and progress for students with specific characteristics could apply to other students with similar characteristics. Researchers have demonstrated that it is possible to identify groups or clusters within students [3, 4], suggesting that it may be possible to use these methods to identify groups of students who could potentially benefit from the same intervention.

There are several published studies that have clustered students into meaningful groups with a goal of driving intervention. Conati et al. provided initial evidence on a user modeling framework for exploratory learning that can automatically identify meaningful student interaction behaviors and can be used to build user models for the online classification of new student behaviors. They built supervised classifiers to recognize categories of student behavior initially identified using an unsupervised clustering approach [5]. Additionally, Rodrigo et al. used unsupervised clustering on data gathered from an intelligent tutoring system to determine whether it was possible to identify distinct groups of students based on interaction logs alone. With the aim of automatic development of detectors of behavior and affect, they identified two student behaviors clusters associated with differing higher-level behaviors and affective states [6]. Another study categorized learners in 13 MOOC courses based upon their interaction with the course, using k-means clustering, to suggest possible improvements in course design and delivery. They identified learners' classes as Uninterested, Casuals, Performers, Explorers and Achievers, where each class of learners had distinct interaction with the course and followed a certain learning approach. Another study has analyzed engagement patterns on four MOOCs and identified two clusters with seven distinct patterns of engagement, suggesting that patterns of engagement in the MOOCs under study were influenced by decisions about pedagogy [7].

In this paper, we use students' characteristics to identify groups of students who could benefit from the similar intervention. Section 2 describes the data and clustering techniques, Sect. 3 shows some exploratory analysis, and Sects. 4 and 5 include the results and discussion.

2 Data

2.1 ALEKS

ALEKS is a Web-based artificially intelligent learning and assessment system. Its artificial intelligence is based on a theoretical framework called Knowledge Space Theory (KST) [8]. KST allows the representation of a large number of knowledge states that constitute a domain in form of a knowledge map. Therefore, KST allows for a precise description of a student's current knowledge state (KS), and what they are ready to learn next. ALEKS's assessment engine enables estimation of students' KS by a diagnostic test taken when the student starts using the system. ALEKS then conducts assessments during students' progress through the course to continuously update students' KS and decide on what the student is ready to learn next. These progress assessments are taken if the student have achieved certain amount of learning progress or have not used the system within 60 days. Research has shown that using ALEKS after school is as effective as interacting with expert instructors [9].

2.2 Data

The data used in this study is from 18 classes in higher education using ALEKS for Beginning Algebra. The data is comprised of information about each assessment the students have taken and has a 5725 total number of assessments for all students. After removing students who did not take the initial assessment, or took it multiple times (making it difficult to estimate their initial knowledge in the class), the dataset has 628 students. These are 16 week classes and we calculate a set of attributes for students up to week 8 in the class. Below attributes are computed for each student:

- Initial assessment score percentage
- Total number of assessments
- Average days between assessments
- Days since class start initial assessment was taken
- Average percentage score increase between assessments.

In addition to above attributes, we have each students' latest assessment score percentage in ALEKS at the end of class.

The above attributes are named based on how ALEKS works and the student behavioral characteristics in ALEKS. Prior knowledge is students' initial knowledge assessment score. Average days between assessments is treated as a measure of students' consistency of working in ALEKS. As mentioned in Sect. 2.1, since each assessment is triggered based on both how much students have learned and/or how much time they've spent, more frequent assessments indicates consistency in learning and time spent. The total number of assessments taken could indicate students' effort in ALEKS, both in learning and time spent. The average increase in percentage of mastery between each assessment indicates how fast/slow the student is mastering the topics and is referred to as pace. These attributes are calculated at/before week 8 (the middle of the course) by filtering the data only for the assessments taken in the first 8 weeks of the class.

3 Exploratory Data Analysis

3.1 Initial Knowledge and Outcome

Students' initial score distribution versus final score distribution is a good indication of how much students have progressed over the class period. Figure 1a shows these distributions on the same plot. It is noticeable that students mostly start with low and medium initial knowledge but the outcome could vary from low to very high scores. Students' progress over the class period can be represented as their outcome minus their initial knowledge. Figure 1b shows how varied students' progress is, over the class period in the same domain on ALEKS.

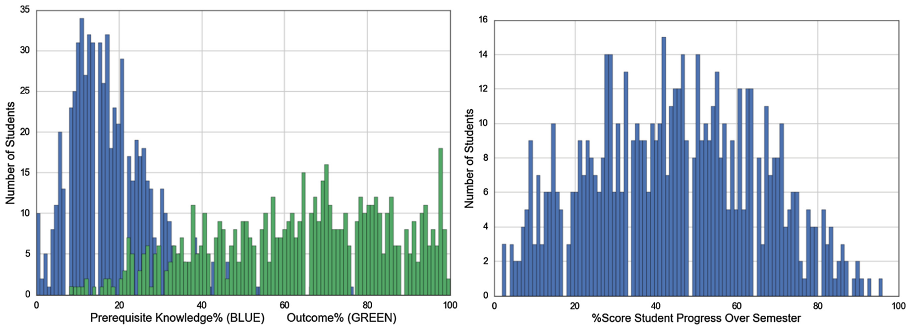


Fig. 1. (a) Students' initial knowledge distribution and outcome distribution. (b) Students' progress over the class period, computed as the difference between students' initial knowledge and outcome in the class.

3.2 Student Progress Versus Pace

Investigating the relationship between students' average items learnt between assessments (progress) and the average number of days between their assessments (pace) reveals that a typical student learns about 18 items over 7 days. Assessments in ALEKS are triggered based on student's learning and time spent. Progress assessments are triggered when a student has learnt 20 topics or spent 10 h in ALEKS. This is shown in Fig. 2, where the highest density is the point at 18 items on the y-axis, representing the average number of items students learnt between assessments, and at 7 days on the x-axis, representing then average days between students' assessments. However, there are students who are considerably faster, learning a given number of items over 5 days or less, and students who are considerably slower, learning the same number of items within 10 days or more.

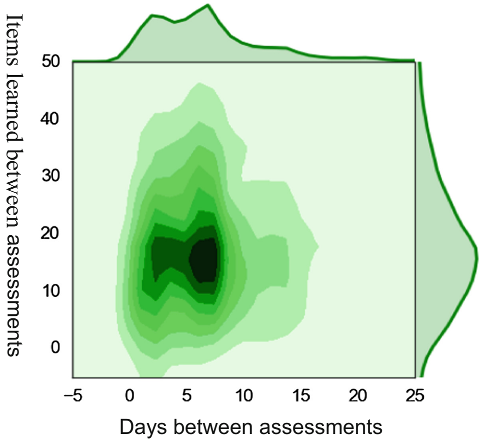


Fig. 2. Relationship between students' average items learned between assessments (progress) and days between assessments (pace).

4 Methodology

We create student profiles within ALEKS using clustering. Specifically, we adopt a three-step process. First, we need to address the high level of correlation between the student characteristics that are input to our clustering algorithm; for example, the Pearson correlation coefficient between last assessment score percentage and total number of assessments for each student is 0.67. To address this concern, we use principal component analysis (PCA), which reduces the data dimensionality by replacing the higher-dimensional original data with a smaller number of non-correlated derived vectors ("principal components"), linear combinations of the original attributes.

Next, we determine the number of clusters using Mean-Shift Clustering [12], which delineates arbitrarily shaped clusters in the data (in this case using a Gaussian kernel) and detects the modes of the density using kernel density estimation. Hence, the number of modes can be used to decide the number of clusters in data by finding the centers of mass within the data.

Finally, we use k-means clustering to find a set of student groups with similar characteristics, choosing this algorithm due to its high interpretability and its property of having non-overlapping clusters. K-means is an iterative clustering technique, where data is partitioned into k clusters by assigning each data point to the cluster with the nearest centroid, and the cluster centroids are iteratively refined until the within-cluster sum of squares (of the distance between each data point and its cluster centroid) cannot be reduced further. In this study, we used Euclidean distance as the distance measure.

5 Results and Discussion

5.1 Preliminary Step: Principal Component Analysis

PCA revealed five principal components (PCs), listed in Table 1. Table 1 shows these factor weightings and the proportion of variance explained by each component. The first three PCs from Table 1 cumulatively explain nearly 90% of variance in the data. Therefore, we use the first three PCs as the input to mean shift clustering and k-means. Note that the first three principal components do not load strongly on the variable "Delay in Start". As such, this variable is effectively not included in the cluster analysis and is not used later in Sect. 5.2 to explain cluster characteristics.

Table 1. Attributes' weights and explained variance proportion for principle components.

PC	Prior knowledge	Consistency	Pace	Effort	Delay in start	Explained variance proportion
1	0.06	0.27	0.29	−0.91	0.00	0.46
2	−0.99	0.00	0.13	−0.02	−0.00	0.31
3	0.04	0.81	0.43	0.38	0.02	0.11
4	−0.10	0.51	−0.84	−0.12	0.02	0.10
5	0.005	0.02	−0.00	0.00	−0.99	0.004

5.2 Learner Profiles

Mean-shift clustering identified 5 possible clusters in the data. Hence, we set k-means clustering to look for five distinct groups of students, using the data from the first three PCs. To interpret these clusters, we then look at the average values for each variable in each cluster, and determine whether each cluster has low, medium or high average values for each variable, in relation to other clusters, shown in Table 2.

Table 2. Average attributes for each cluster.

Label	Size	Prior knowledge (% score)	Consisteny (days)	Pace (% score increase)	Effort (# of assessments)
1	190	13.6 (Very Low)	9.3 (Average)	8 (Average)	5.1 (Low)
2	243	17.8 (Average)	8 (Average)	6.9 (Average)	9.5 (Average)
3	50	15.6 (Average)	23.3 (Very Low)	12.9 (High)	4 (Very Low)
4	62	18 (Average)	5.2 (High)	5 (Low)	16.5 (Very High)
5	83	40 (Very High)	10.2 (Average)	6.8 (Average)	6.5 (Low)

We use t-tests to measure the statistical significance of the difference in each characteristic between clusters, using a Benjamini and Hochberg post-hoc correction [10] to control for running multiple comparisons. Table 3 shows the statistical significance (p-value) of the difference between each attribute between each pair of clusters. All differences between groups that have $p < 0.05$, indicated in bold, are also statistically significant after a Benjamini and Hochberg post-hoc correction, except for the prior knowledge difference between clusters 2 and 3 ($p = 0.04$). As Table 3 shows, each pair of clusters is different in at least three of the four variables.

Table 3. Statistical significance (p-value) of the difference between each attribute between each pair of clusters.

Clst 1	Clst 2	Prior knowledge (% score)	Consistency (days)	Pace (% score increase)	Effort (# of assessments)
1	2	**<.001**	**<.001**	**<.001**	**<.001**
1	3	0.089	**<.001**	**<.001**	**<.001**
1	4	**<.001**	**<.001**	**<.001**	**<.001**
1	5	**<.001**	0.12	**<.001**	**<.001**
2	3	0.04	**<.001**	**<.001**	**<.001**
2	4	0.88	**<.001**	**<.001**	**<.001**
2	5	**<.001**	**<.001**	0.73	**<.001**
3	4	0.20	**<.001**	**<.001**	**<.001**
3	5	**<.001**	**<.001**	**<.001**	**<.001**

5.3 Learner Profile Names

Using the cluster characteristics in Table 2, we have identified each learner profile with a name. Below are the cluster names and description of each profiles.

1. Strugglers: this group starts with a very low prior knowledge, puts in low effort and has an average pace of learning.
2. Average Students: this group of learners are average in all characteristics.
3. Sprinters: this group starts with average prior knowledge. They have low consistency in learning and low effort, but have a high pace.
4. Gritty: this group has an average prior knowledge. They have high consistency and high effort, but work at a slow and steady pace.
5. Coasters: this group starts with very high prior knowledge. However, they have average pace and consistency, and put in low effort.

The name of the gritty group is inspired by Angela Duckworth's definition of grit as perseverance and passion to achieve long-term goals [11]. This group shows similar characteristics to what Duckworth defines as grit, maintaining consistency and high effort throughout the class.

A good way of understanding these clusters comes from an external qualitative study conducted by an independent research agency to identify three key student personas in higher education. These personas were characterized based on organizational effort, study effort, persistence, self-confidence, and social extroversion:

- The struggler: characterized by low effort, persistence, self-confidence, social extroversion and very high organizational efforts.
- The planner: characterized by very high organizational and study effort in addition to high persistence, self-confidence and social extroversion.
- The average student: characterized by very low organizational effort, low study effort and persistence, and average self-confidence and social extroversion.

These personas characterize a converging set of categories that represent one or more groups of students we identified in our current cohort of students. The planner characteristics are very similar to gritty students. The struggler persona represents the strugglers and sprinters in our study while the average student persona covers the characteristics of average Students and coasters.

5.4 Learner Profiles and Mastery

We can use final percentage mastery in ALEKS to verify whether the profiles are associated with differences in students' outcomes at the end of the class. Based on their learning characteristics in the first 8 weeks of the class, we expect the strugglers to achieve a low outcome, Average Students to achieve an average outcome, Sprinters to achieve an average to low outcome, the Gritty group to achieve high outcome and Coasters to achieve an average to high outcome.

Table 4 shows the final percentage mastery for each group and the corresponding standard deviation. As we can see, the groups differ in their mastery in the hypothesized fashions. For example, gritty learners finish the class with a very high outcome in

ALEKS while strugglers are at the other end. We then conduct a set of t-tests to measure the statistical significance differences in final mastery between groups, using a Benjamini and Hochberg post-hoc correction [10] to control for running multiple comparisons. Table 5 shows the mean differences between different groups and their statistical significances. All differences between groups that have $p < 0.05$ are also statistically significant after a Benjamini and Hochberg post-hoc correction. We find that Groups 1 and 3 (Strugglers and Sprinters) and groups 2 and 5 (Average Students and Coasters) have similar average final mastery. What differentiates these groups is their characteristics in terms of consistency, pace and effort.

Table 4. Final percentage mastery and corresponding standard deviation for each profile.

Label	Learner profile	% Final mastery	Standard deviation	% Final mastery
1	Strugglers	44.8	17.9	Very low
2	Average students	72.4	15.4	Average
3	Sprinters	48.9	19.5	Low
4	Gritty	88.6	11.8	Very high
5	Coasters	72.7	15.1	Average

Table 5. Mean difference between different groups and their statistical significance. All differences between groups that have $p < 0.05$ are also statistically significant after a Benjamini and Hochberg post-hoc correction.

Group	Comparison group	Mean difference in outcome	P-value
1	2	−27.6	<.001
1	3	−4.1	0.15
1	4	−43.8	<.001
1	5	−27.9	<.001
2	3	23.5	<.001
2	4	−16.2	<.001
2	5	−0.3	0.86
3	4	−39.7	<.001
3	5	−23.8	<.001
4	5	15.9	<.001

6 Discussion and Conclusion

In this study, we aimed to address the challenge of deciding and applying individual interventions for students by identifying groups of students whose learning patterns were similar, and who could potentially benefit from the same intervention. We used cluster analysis techniques to identify groups of students with similar academic and behavior characteristics who can benefit from the same intervention. To accomplish this, we used PCA and two clustering techniques to identify distinct groups of students within 628 students using ALEKS. Our analyses used five characteristics of these students up until the midterm of the class: students' starting score in the course (prior knowledge), average score increase between assessments (progress), total number of assessments taken (effort), average days between assessments (pace), and days since class start that initial assessment was taken (delay in start). We find five clusters, described in Table 2 in terms of their average characteristics. Translating these averages to word descriptions gives us the learning pattern of students in each cluster (Table 3), producing five student profiles – Strugglers, Average Students, Sprinters, Gritty, and Coasters – which can be communicated to instructors. The clusters can then be used by instructors to devise appropriate interventions in time to still take meaningful action – at the course's midterm. Used appropriately, with the proper interventions, these profiles can ensure that institutions are effectively using the information from adaptive learning systems such as ALEKS to deliver relevant learner intervention. For example, the Strugglers group could benefit from extra instruction while the Sprinters group could benefit from nudges from the learning platform or instructor to maintain learning consistency and put in higher efforts [12].

The next step for this project is to devise and implement interventions using the guidelines given in this study and study the effectiveness of these interventions on individuals within each group, to understand the effectiveness of this grouping for intervention. Another area of future work would be to identify how early each student's learning group could be identified to enable the instructors to intervene early in the course – if a cluster could be identified earlier, it would enable faster and possibly more effective intervention. In addition, learner profiles could be validated further using relevant surveys at the beginning of the semester, before students start using ALEKS. For example, we may be able to use the short grit survey [16], to see if students who self-report as gritty tend to also behave in a gritty fashion within ALEKS.

Through these approaches, we can better understand the learner profiles we are developing and use them to improve student outcomes, helping adaptive learning systems to achieve their goals for helping every student succeed.

Acknowledgements. This paper is based on work supported by the McGraw-Hill Education Digital Platform Group. We would like to extend our appreciation for all the informational support provided by ALEKS team at McGraw-Hill Education, specially Jeff Matayoshi, applied research scientist, and Eric Cosyn, director of applied research. Any opinions, findings, conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect positions or policies of the company.

References

1. Lin-siegler, X., Dweck, C.S., Cohen, G.L.: Instructional interventions that motivate classroom learning. J. Educ. Psychol. **108**(3), 295–299 (2016)
2. Paunesku, D., Walton, G.M., Romero, C., Smith, E.N., Yeager, D.S., Dweck, C.S.: Mind-set interventions are a scalable treatment for academic underachievement. Psychol. Sci. **26**(6), 784–793 (2015)
3. Bouchet, F., Harley, J.M., Trevors, G.J., Azevedo, R.: Clustering and profiling students according to their interactions with an intelligent tutoring system fostering self-regulated learning. JEDM - J. Educ. Data Min. **5**(1), 104–146 (2013)

4. Beal, C.R., Qu, L., Lee, H.: Classifying learner engagement through integration of multiple data sources. In: Proceedings of the National Conference on Artificial Intelligence, vol. 21, no. 1, p. 151 (2006)
5. Amershi, S., Conati, C.C.: Combining unsupervised and supervised classification to build user models for exploratory. JEDM - J. Educ. Data Min. **1**(1), 1–54 (2009)
6. Rodrigo, M.M.T., Anglo, E.A., Sugay, J.O., Baker, R.S.J.D.: Use of unsupervised clustering to characterize learner behaviors and affective states while using an intelligent tutoring system. In: International Conference on Computers in Education (2008)
7. Ferguson, R., Clow, D.: Examining engagement: analysing learner subpopulations in massive open online courses (MOOCs). In: Proceedings of the Fifth International Conference on Learning Analytics and Knowledge - LAK 2015, pp. 51–58 (2015)
8. Falmagne, J.-C., Cosyn, E., Doignon, J.-P., Thiéry, N.: The Assessment of Knowledge, in Theory and in Practice. In: Missaoui, R., Schmidt, J. (eds.) ICFCA 2006. LNCS (LNAI), vol. 3874, pp. 61–79. Springer, Heidelberg (2006). https://doi.org/10.1007/11671404_4
9. Craig, S.D., et al.: The impact of a technology-based mathematics after-school program using ALEKS on student's knowledge and behaviors. Comput. Educ. **68**, 495–504 (2013)
10. Benjamini, Y., Hochberg, Y.: Controlling the false discovery rate: a practical and powerful approach to multiple testing. J. R. Stat. Soc. Series B (Methodol.) **57**, 289–300 (1995). WileyRoyal Statistical Society
11. Duckworth, A.L., Peterson, C., Matthews, M.D., Kelly, D.R.: Grit: perseverance and passion for long-term goals. J. Pers. Soc. Psychol. **92**(6), 1087–1101 (2007)
12. Arnold, K.E., Pistilli, M.D., Arnold, K.E.: Course signals at purdue: using learning analytics to increase student success. In: 2nd International Conference on Learning Analytics Knowledge, pp. 2–5, May 2012

Identifying How Metacognitive Judgments Influence Student Performance During Learning with MetaTutorIVH

Nicholas V. Mudrick[1](✉), Robert Sawyer[2], Megan J. Price[1], James Lester[2], Candice Roberts[3], and Roger Azevedo[1]

[1] Department of Psychology, North Carolina State University, Raleigh, NC, USA
{nvmudric,mjprice3,razeved}@ncsu.edu

[2] Department of Computer Science, North Carolina State University, Raleigh, NC, USA
{rssawyer,lester}@ncsu.edu

[3] Natural Sciences Department, Wake Technical Community College, Raleigh, NC, USA

Abstract. Students need to accurately monitor and judge the difficulty of learning materials to effectively self-regulate their learning with advanced learning technologies such as intelligent tutoring systems (ITSs), including MetaTutorIVH. However, there is a paucity of research examining how metacognitive monitoring processes such as ease of learning (EOLs) judgments can be used to provide adaptive scaffolding and predict student performance during learning ITSs. In this paper, we report on a study investigating how students' EOL judgments can influence their performance and significantly predict their learning outcomes during learning with MetaTutorIVH, an ITS for human physiology. The results have important design implications for incorporating different types of metacognitive judgements in student models to support metacognition and foster learning of complex ITSs.

Keywords: Metacognitive monitoring · Ease of learning judgments
Performance · Predictive modeling · Intelligent tutoring systems

1 Introduction

The use of advanced learning technologies, such as intelligent tutoring systems (ITS), for learning is becoming ubiquitous and students learning with these environments are expected to act autonomously and self-regulate their learning [1]. Furthermore, several ITSs, such as MetaTutor, Betty's Brain, AutoTutor, and nSTUDY have been developed to detect, support, and foster students' metacognition and self-regulated learning (SRL) [2]. As such, it is imperative for students to accurately monitor the difficulty of the material they learn with these environments. These judgments regarding how difficult content will be to learn, or ease of learning (EOL) judgments, are important contributors to academic achievement and learning with ITSs as they can influence attention, time, strategy use, and effort allocation to the learning content [1–3]. Past research investigating these judgments has assessed their accuracy and confidence (i.e., difference between their judged and demonstrated levels of performance) and has found in most

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 140–149, 2018.
https://doi.org/10.1007/978-3-319-91464-0_14

cases, students are largely inaccurate and overconfident [1–4]. However, this research has primarily investigated EOLs in laboratory-based contexts (i.e., paired-associates learning) that may not reflect how students make these judgments in educational contexts. Thus, determining the utility of EOL judgments in predicting learning outcomes provides a valuable research contribution to using metacognitive judgment features in student modeling during learning with ITSs.

1.1 Related Work

EOL judgments can contribute to successful learning outcomes because they are made early on in the learning process and can influence study behavior and allocation of effort during self-regulated learning [1–3]. Research examining these judgments in laboratory settings suggests that EOLs are poor to moderate predictors of learning outcomes because they are prospective judgments that are made without seeing the instructional materials. However, much of this literature examines how EOL judgments can influence learning with simple tasks (i.e., paired-associates learning) and as such, most of the factors that influence the accuracy of these judgments are relatively micro-level (e.g., semantic relatedness between word pairs, fluency of perceptional processing, [2]). Although the results from laboratory studies have provided evidence regarding the inability of EOLs to predict performance, there is a potential for using more multimedia instructional materials (e.g., text and diagrams), to identify how EOLs can be embedded within ITSs to predict student performance during complex learning.

To explore this potential, we investigate how micro-level metacognitive judgments (EOL judgments) that are made after examining a science question can influence student performance. EOLs may force a learner to activate and successfully retrieve relevant prior knowledge (if any), plan and generate sub-goals for learning based on successful retrieval of relevant prior knowledge, to prepare to actively and accurately monitoring and regulate their cognition (e.g., select learning strategies), motivation (e.g., expect to persist given the complexity of the materials and lack of relevant prior knowledge), and emotions (e.g., engage in cognitive reappraisal when experiencing frequent and prolonged bouts of confusion and frustration). These are possible cognitive, affective, motivation, and metacognitive SRL processes that can be activated prior to an EOL that may fluctuate once the multimedia material is made available by an ITS and can therefore substantially influence and predict students' performance, especially in those with low prior knowledge such as the ones who participated in our study (see Sect. 2.1).

2 Current Study

To assess the relationship between students' EOL judgments and student performance during learning with MetaTutorIVH, we investigated the following research questions:

1. Are there differences in performance when students judge the content as easier to learn vs. when students judge the content as more difficult to learn?

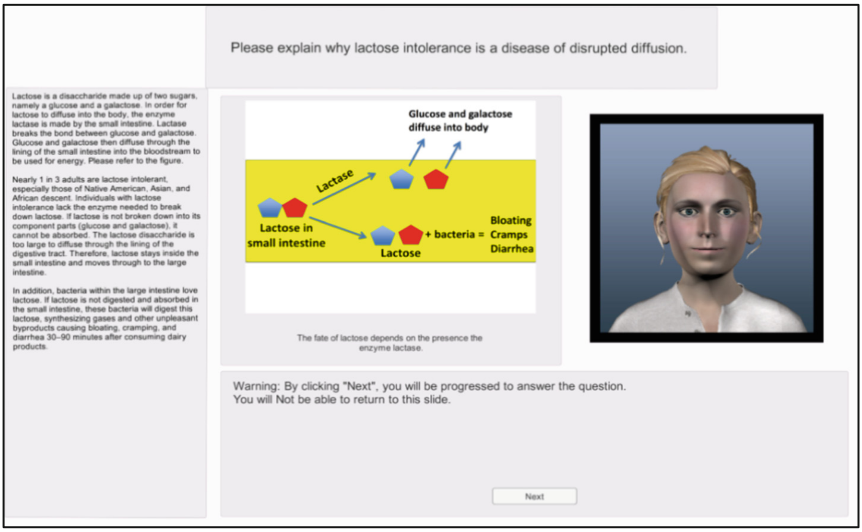
2. Are there differences in performance for when students judge the content as easier to learn *than it actually is* vs. when students judge the content as more difficult to learn *than it actually is*?
3. Can students' ease of learning (EOL) judgments predict their performance?

2.1 Participants

48 undergraduate students (77% female) enrolled at a large mid-Atlantic North American University participated in this study. Their ages ranged from 18 to 30 (*M* = 20.30, *SD* = 2.35). Scores from the 18-item pre-test assessing their prior knowledge of the science domains covered in the study revealed that students had low to moderate prior knowledge of the science content (*M* = 11.20 [62.22%], *SD* = 1.48 [8.22%]). Students were monetarily compensated up to $30 dollars for their participation.

2.2 MetaTutorIVH

The study was conducted with MetaTutorIVH, where students made several metacognitive judgments, inspected multimedia materials, and answered a series of multiple-choice questions regarding 9 different human physiological systems (e.g., circulatory, endocrine, nervous, etc.). MetaTutorIVH was designed to examine the influence of an intelligent virtual human's (IVH) behavior on students' cognitive learning strategies, metacognitive judgments, and emotions during learning about complex biology topics (Fig. 1). The environment consists of an IVH, text passages and diagrams about human body systems, and metacognitive judgment prompts. For this study, the IVH's behavior



Please explain why lactose intolerance is a disease of disrupted diffusion.

Lactose is a disaccharide made up of two sugars, namely a glucose and a galactose. In order for lactose to diffuse into the body, the enzyme lactase is made by the small intestine. Lactase breaks the bond between glucose and galactose. Glucose and galactose then diffuse through the lining of the small intestine into the bloodstream to be used for energy. Please refer to the figure.

Nearly 1 in 3 adults are lactose intolerant, especially those of Native American, Asian, and African descent. Individuals with lactose intolerance lack the enzyme needed to break down lactose. If lactose is not broken down into its component parts (glucose and galactose), it cannot be absorbed. The lactose disaccharide is too large to diffuse through the lining of the digestive tract. Therefore, lactose stays inside the small intestine and moves through to the large intestine.

In addition, bacteria within the large intestine love lactose. If lactose is not digested and absorbed in the small intestine, these bacteria will digest this lactose, synthesizing gases and other unpleasant byproducts causing bloating, cramping, and diarrhea 30–90 minutes after consuming dairy products.

The fate of lactose depends on the presence the enzyme lactase.

Warning: By clicking "Next", you will be progressed to answer the question.
You will Not be able to return to this slide.

Glucose and galactose diffuse into body
Lactase
Lactose in small intestine
Lactose
+ bacteria =
Bloating
Cramps
Diarrhea
Next

Fig. 1. Screenshot of MetaTutorIVH's main interface illustrating the science questions, multimedia content, and intelligent virtual human (IVH).

consisted of specific facial expressions that were dependent on the relevancy of the content (see Research Design in Sect. 2.3).

Students interacted with MetaTutorIVH over 18 counter-balanced, randomized, self-paced trials that consisted of science questions, metacognitive judgment prompts, multimedia science content, and multiple-choice questions. The 18 trials were identical in format. In each trial, students were first presented with a science question regarding a particular body system on a separate slide before being presented with the multimedia science content. An example science question was, "*Please explain the process by which we inhale more oxygen molecules than we exhale.*" After viewing the science question, students were then asked to submit an EOL judgment by answering, "*How easy do you think it will be to learn the information needed to answer this question?*" Students submitted their responses on a 0–100% scale, increasing in increments of 1%.

Following the submission of their EOL judgment, students were presented with a content page containing the text passage, diagram depicting the concept described in the text, the IVH, and the science question that was presented previously. After 30 s, students were prompted to judge the relevancy of the text and diagram to the science question they needed to answer by responding on a 3-point Likert-style scale. After students made their text and diagram content evaluations, the IVH facially expressed a congruent, incongruent, or neutral facial expression depending on the content relevance. Students returned to reading the text and inspecting the diagram. After they were finished viewing the content, students were required to answer the science question they were presented previously by choosing a correct response from 4 options. After they submitted their answer, students were prompted to make a judgment assessing their confidence in their chosen answer. After they submitted their judgment, students were prompted to justify their answer by typing a response into a text box (to ensure they had not skimmed the material and guessed). They were then asked to make another confidence judgment based on their justification. This procedure was repeated for the remainder of the 18 trials following the experimental session.

2.3 Research Design

This study used a 3 × 3 × 2 within-subjects design resulting in 18 trials. The first factor was content relevancy, which referred to the relationship between the level of description of the concept presented in the text/diagram to the science question asked. Students interacted with 3 levels of relevance: high relevance (where both the text and diagram were fully relevant to the science question asked), low text relevance (where the diagram was fully relevant to the science question, but the text depicted the science topic in more general terms), and low diagram relevance (where the text was fully relevant to the science question, but the diagram depicted the science topic more generally). Despite the presence of less relevant text or diagrams, the content still contained the information needed to correctly answer the question. The second factor was the congruency of the IVH's facial expressions such that the IVH facially expressed a congruent (i.e. the facial expression matched the relevancy of the content, joy for fully relevant content, confusion for less relevant), incongruent (i.e. the facial expression did not match the relevancy of the content, confusion for fully relevant content, and joy for less relevant content), or a neutral (included as a comparison) based on the relevancy of the content. For example, if the text was only somewhat relevant to the content, the IVH facially expressed confusion to be congruent with the content, or joy to be incongruent with the content. The third factor was whether the science question asked about a standard function or malfunction of a particular body system. For example, a function question about the human respiratory system was, "*Please explain the process by which we inhale more oxygen molecules than we exhale*," while a malfunction question was, "*Please explain how, in cystic fibrosis patients, a missing chloride channel alters diffusion of oxygen in the respiratory system*."

2.4 Materials and Procedure

The study materials and equipment included the following: demographics questionnaire, pretest made of 18 4-option multiple-choice questions used to assess prior knowledge of the body systems described within the environment. Students EOL judgments and answers to the 4-option multiple-choice questions were automatically collected by MetaTutorIVH.

Students completed an informed consent form and then asked to complete a computerized demographic questionnaire and an 18-item science content pretest assessing their basic biology content knowledge. After students completed the pretest, they completed the 18 previously described trials with MetaTutorIVH. The average interaction with MetaTutorIVH lasted approximately 1 h (M = 58.5 m, SD =20.40 m).

2.5 Data Sources and Preprocessing

Traditionally, metacognitive confidence judgments like EOL judgments have been examined by calculating their absolute and relative accuracies. Absolute accuracy is defined as the difference between the judgment and performance, while relative accuracy is the relationship between a set of judgments and students' performance scores. While absolute accuracy identifies how precise a student is in their metacognitive judgments, relative accuracy assesses the correspondence between judgments and overall performance [1, 4]. However, to assess relative and absolute accuracies, judgments and performance measures must be on the same continuous or ordinal scale, which may not reflect the types of assessments used in academic settings (e.g., multiple-choice questions, ordinally graded essays, etc.) or in this study. As such, a standardization process on students' EOL judgments was performed.

Although the experimental manipulations in this study included changes to the content relevancy to the science question, as well as the behavior of the IVH, students did not have access to the text, diagram or agent when making their EOL judgment for a given trial. An F-test on the significance of the coefficients for content relevancy, IVH facial expression congruence, and type of science question in a multiple linear regression for content difficulty (see below) indicated none of the coefficients were significantly different from zero ($F(5, 12) = 0.63$, $p = .68$). As such, results suggest these factors did not significantly affect the difficulty of the content.

Standardizing EOL Judgments by Student. We standardized students' EOL judgment scores *by student*. The resulting standardized measures of a student's EOL represent how easy the student believed the multiple-choice problem to be relative to other problems the student had rated. For example, a standardized value of 1.5 meant that the student believed this problem was 1.5 standard deviations easier than their average EOL judgment. A negative value indicated that the student believed the problem was harder than their average EOL judgment.

This type of standardization is important because students could have had different interpretations of a problem being "easy" or "difficult", which is reflected of their ratings on the 0–100 scale. For example, on the original 0–100 scale, one student's average EOL judgment was 17.7, while another's was 76.4. These two students demonstrate different interpretations of the 0–100 scale for their EOL judgments. This is important, considering that the first student, who thought the content was substantially more difficult to learn according to their raw EOL judgment values, correctly answered 12 of 18 (66.7%) questions while the second only correctly answered 9 of 18 (50%) questions. As such, this standardization allowed comparisons between students since the standardized values represent a student's relative EOL.

Assessing Content Difficulty. Each trial included a multiple-choice question with four possible responses. For each multiple-choice question, there was one correct response, two partially correct responses, and one incorrect response. We calculated a weighted sum of a questions' ease from the total number of students who answered the multiple-choice questions according to these three response categories (i.e., correct = 1, partially correct = 0.5, incorrect = 0). Higher weighted sums indicate that more students answered correctly or partially correct (indicating easier to learn content), while lower weighted sums indicate that more students answered incorrectly or partially correct (indicating more difficult to learn content). These weighted sums were calculated for each of the 18 trials, where each trial's weighted sum ranged from 0 to the total number of students who responded to those questions.

Determining EOL Judgment Error. Standardizing students' EOL judgments allowed us to assess the accuracy of their EOL judgments against the measure of content difficulty. Specifically, we calculated accuracy in terms of students' EOL judgment error, in contrast to traditional measures of relative and absolute accuracies. There were two error measures of interest: *signed error* and *squared error.*

We calculated the *signed error* as the difference between a student's standardized EOL and the standardized problem difficulty (similar to the traditional measure of relative accuracy). Because the resulting value retained its positive or negative value, this allowed us to assess students' EOL judgments in comparison to the actual difficulty of the problem. For example, if the signed error was positive, the student thought the problem was easier *than it actually was*, whereas if the signed error was negative, the student thought the problem was more difficult *than it actually was.*

The *squared error* was calculated as the signed error value squared. This allowed us to calculate of the magnitude of error in a student's EOL judgment the problem difficulty (similar to traditional measures of absolute accuracy). This measure was also motivated by noting that the sum of squares is a common regression error function.

We calculated both of these error measures on a student-trial basis such that our units of analysis were students' EOL judgments per question, as opposed to aggregating these judgments by student. As such, each time a student made a judgment, the error was calculated, for a total of 18 signed and squared errors per student (i.e., 18 trials × 48 students = 864 signed error values, and 18 × 48 = 864 squared error values).

3 Results

3.1 Are There Differences in Performance When Students Judge the Content as Easier to Learn vs. When Students Judge the Content as More Difficult to Learn?

The student-standardized EOL judgments were used to determine whether there were differences in performance per trial for students who judged the content as easier to learn vs. those who judged the content as more difficult to learn. More specifically, a one-way ANOVA was conducted to compare the effect of positive (judged to be easier to learn than average) or negative (judged to be harder to learn than average) student standardized EOL judgments on performance. There were 438 (50.6%) trials in which students judged the content to be easier to learn than their average EOL for the content and 426 (49.4%) trials in which students judged the content to be more difficult to learn than their average EOL judgment. There was a significant effect of students' EOL judgments on performance at the $\alpha = 0.05$ level for the positive (easier to learn) or negative (harder to learn) judgment groupings [$F(1, 862) = 4.02, p = 0.045$]. A post-hoc analysis using a Welch's (unequal variance) two sample t-test indicated that the performance for students who judged the content as easier to learn than average ($M = 0.74, SD = 0.34$) was significantly *lower* than students who judged the content as more difficult to learn than average ($M = 0.78, SD = 0.32$). Furthermore, a significant negative correlation ($r = -.10, p = 0.003$) was observed between the standardized student EOLs and performance. This demonstrated that as students judged the content to be easier to learn relative to other content, the worse they performed on the multiple-choice questions. As such, results indicated that students who judged the content as more difficult to learn achieved higher performance on the multiple-choice questions.

3.2 Are There Differences in Performance for When Students Judge the Content as Easier to Learn *Than It Actually Is* vs. When Students Judge the Content as More Difficult to Learn *Than It Actually Is*?

The signed errors of EOL judgments were used to determine if there were differences in performance per trial among students who judged the content as easier to learn *than it actually was* vs. performance among students who judged the content as more difficult to learn *than it actually was*. A one-way ANOVA was conducted to compare the effect of positive (i.e., judged as easier to learn than it was) or negative (i.e., judged as harder to learn than it was) signed error judgement groupings on performance. There were 422

(48.8%) trials in which students judged the content as easier to learn than it actually was, and 442 (51.2%) trials in which students believed the content to be harder to learn than it actually was. There was a significant effect of the signed judgment error on performance at the $\alpha = 0.05$ level for the positive and negative judgment groupings [$F(1, 862) = 101.3$, $p < 0.001$]. A post-hoc analysis using a Welch's two sample t-test indicated that performance for students who judged the content as easier than it actually was ($M = 0.65$, $SD = 0.36$) was significantly lower than students who judged the content as more difficult to learn than it actually was ($M = 0.86$, $SD = 0.26$). Additionally, a significant negative correlation between the signed error and performance was observed ($r = -.37$, $p < .001$), such that as students who judged the content to be easier to learn than it actually was, the worse they performed on the multiple-choice question. As such, results indicated that students who judged the content as more difficult to learn than it actually was achieved higher performance on the multiple-choice questions (Table 1).

Table 1. Summary of multiple choice score by EOL grouping, which summarizes research questions 1 (top row) and 2 (bottom row).

	Positive group			Negative group			Overall correlation
	n	*M*	*SD*	*n*	*M*	*SD*	*r* (p)
Standardized EOL	438	0.74	0.34	426	0.78	0.32	−0.1*
Signed EOL Error	422	0.65	0.36	442	0.86	0.26	−0.37**

*$p < .05$. **$p < .001$.

3.3 Can We Use Ease of Learning (EOL) Judgments to Predict Student Performance?

For these analyses, we treated the performance prediction as a binary classification problem. This was done by treating partially correct answers as incorrect, resulting in a 61.5% performance correctness rate serving as the majority class (question answered correctly) baseline. We computed the 10-fold cross validation accuracy using a multilayer perceptron model with layers of 15 and 5 rectified linear units implemented from the sklearn.neural_network package in Python [6] and using standardized EOL judgments, difficulty direction correct, and whether the standardized ease of learning is positive as features. The difficulty direction correct is a binary variable indicating whether the standardized EOL judgment and standardized ease of content have the same sign. This is an indicator of whether or not the student correctly assessed the content of being more or less difficult to learn than the average content and was correctly performed on 47.5% of trials. These predictors were chosen because they are almost independent of the difficulty of the content and reflect the usefulness of the student's EOLs judgments without explicitly including the content's difficulty. The three predictors used as input features, including the content's difficulty used to calculate the errors, are calculated and standardized using only data from the training fold. The average accuracy across the 10-folds was 71.7%, which is a significant improvement over the majority class baseline of 61.5% ($t = 5.88$, $p < 0.001$). This accuracy improvement indicates that these features based primarily upon student EOL judgments are useful in predicting student performance.

4 Discussion

The study investigated how students' EOL judgments can influence and be predictive of performance. The results from these analyses significantly augment our understanding of how students' metacognitive judgments can be used to model performance during learning with ITSs and have implications designing future ITSs that emphasize the role of metacognition during complex learning.

Results from our first research question indicated that students who judged the content as being harder to learn outperformed students who judged the content as being easier to learn on their multiple-choice responses. These findings are compounded by results from research question 2, which indicated that students who judged the content as being harder to learn than it actually was, significantly outperformed students who judged the content to be easier to learn than it actually was. The significantly lower performance of students who judged the content to be easy suggests that students' overconfidence for these questions deleteriously impacted their performance. Contrary to published literature on EOLs, it is possible that these students did not accurately monitor their emerging understanding, select the appropriate cognitive strategies, and allocate sufficient effort necessary to successfully understand, and learn the multimedia materials, leading to poor performance [1–4]. Alternatively, students who had judged the content as being harder to learn achieved superior performance by accurately monitoring their understanding and selecting the appropriate strategies, allocating more effort than they needed to successfully understand the content. These results significantly extend previous research on EOL judgments by integrating different types of measures to indicate relative and absolute judgment accuracies (i.e., standardizing both EOL judgments and problem difficulty). Traditionally, research has addressed these measures of accuracy separately by calculating the absolute accuracy index for absolute accuracy and Goodman-Kruskal correlations for relative accuracy (see [5]).

Lastly, results from our third research question demonstrated the utility of EOLs in predicting student performance. Results indicated that including EOL judgments and their accuracy as predictors in a multi-layer perceptron model demonstrated statistically significant predictions of performance 71.7% of the time, improving prediction by 10.2% (relative improvement of 16.6% over the baseline). As such, it is possible that the context of learning with educationally relevant materials may facilitate more accurate EOL judgments than the other contexts where they have been examined. Furthermore, limited research has investigated including metacognitive judgments as features to use while building accurate student models. Therefore, our results provide evidence that prospective metacognitive judgments can provide ITSs with important student-based performance information with which the system can use to identify the accuracy of metacognitive monitoring processes during learning and intervene accordingly based on a sophisticated student model.

From practical and design perspectives, incorporating EOL-like features into ITSs is straightforward and imposes little burden on students. An ITS need only take one continuous input from a brief preview of a future problem, without showing the student any content, to predict their performance for learning that content. Results from our analyses indicated that students generally spent little time making these judgments (M = 4.5 s, SD = 3.3 s) relative to their overall time interacting with MetaTutorIVH in our study (average of 2.3% of the total time). As such, integrating this as a feature in ITSs can potentially provide the system pertinent performance information. For example, students could provide an EOL after being presented with their next topic. The ITS uses that students' EOL judgment to model their performance and intervenes based on this prediction. Specifically, the IVH or other artificial agent (knowledgeable of the difficulty of the content) could provide scaffolding to the student in the form of suggestions to re-evaluate their metacognitive judgment, slow down and pay attention to the upcoming content, etc. Alternatively, the IVH could then prompt the student to engage in context-appropriate cognitive learning strategies by having the students summarize the presented material, make inferences about the content, and integrate the information in the text and diagrams to facilitate better conceptual understanding and deeper learning.

Acknowledgements. This research was supported by funding from the National Science Foundation (DRL#1431552). The authors would also like to thank members from the SMART Lab and IntelliMedia Group for their contributions to this project.

References

1. Azevedo, R., Taub, M., Mudrick, N.: Understanding and reasoning about real-tie cognitive, affective, metacognitive processes to foster self-regulation with advanced learning technologies. In: Handbook of Self-Regulation of Learning and Performance. Routledge, New York (2018)
2. Dunlosky, J., Metcalfe, J.: Metacognition: A Textbook for Cognitive, Educational, Life Span and Applied Psychology. SAGE, Newbury Park (2009)
3. Azevedo, R., Mudrick, N.V., Taub, M., Bradbury, A.: Self-regulation in computer-assisted learning systems. In: Handbook of cognition and education (In press)
4. Feyzi-Behnagh, R., Azevedo, R., Legowski, E., Reitmeyer, K., Tseytlin, E., Crowley, R.S.: Metacognitive scaffolds improve self-judgments of accuracy in a medical intelligent tutoring system. Instr. Sci. **42**, 159–181 (2014)
5. Schraw, G.: A conceptual analysis of five measures of metacognitive monitoring. Metacogn. Learn. **4**, 33–45 (2009)
6. Pedregosa, F., Varaquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)

Predicting Learners' Emotions in Mobile MOOC Learning via a Multimodal Intelligent Tutor

Phuong Pham and Jingtao Wang(✉)

Computer Science and LRDC, University of Pittsburgh, Pittsburgh, PA, USA
{phuongpham,jingtaow}@cs.pitt.edu

Abstract. Massive Open Online Courses (MOOCs) are a promising approach for scalable knowledge dissemination. However, they also face major challenges such as low engagement, low retention rate, and lack of personalization. We propose AttentiveLearner^2, a multimodal intelligent tutor running on unmodified smartphones, to supplement today's clickstream-based learning analytics for MOOCs. AttentiveLearner^2 uses both the front and back cameras of a smartphone as two complementary and fine-grained feedback channels in real time: the back camera monitors learners' photoplethysmography (PPG) signals and the front camera tracks their facial expressions during MOOC learning. AttentiveLearner^2 implicitly infers learners' affective and cognitive states during learning from their PPG signals and facial expressions. Through a 26-participant user study, we found that: (1) AttentiveLearner^2 can detect 6 emotions in mobile MOOC learning reliably with high accuracy (average accuracy = 84.4%); (2) the detected emotions can predict learning outcomes (best $R^2 = 50.6\%$); and (3) it is feasible to track both PPG signals and facial expressions in real time in a scalable manner on today's unmodified smartphones.

Keywords: Heart rate · Facial expression · Multimodal interface
Massive Open Online Course · Intelligent Tutoring System · Affective computing
Mobile device

1 Introduction

Despite the popularity and rapid growth, current Massive Open Online Courses (MOOCs) still have much higher in-session dropout rates (e.g. 55.2% in [10]) and lower completion rates (e.g. 7.7% [1]) when compared with similar courses offered in traditional classrooms. In addition to apparent disadvantages such as increased external distractions [22], passive video-watching experiences, and lack of sustained motivations to study alone [14], the limited information exchange between instructors and learners can be another crucial factor restricting the efficacy of MOOCs. Whereas previous work [5, 8] on Intelligent Tutoring Systems (ITSs) showed the feasibility of analyzing the learning process from students' cognitive and affective states via various sources of information, e.g. physiological signals [8] or facial expressions [5], most of the previous

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 150–159, 2018.
https://doi.org/10.1007/978-3-319-91464-0_15

approaches require additional sensing hardware. These additional requirements could be an obstacle to learning at scale due to their extra costs, availability, and limited portability.

In response to these challenges, we propose AttentiveLearner^2 (read as "*attentive learner squared*") [17], a multimodal intelligent MOOC tutor running on unmodified smartphones (Fig. 1). AttentiveLearner^2 uses on-lens finger gestures to control video playback (i.e. covering and holding the back-camera lens to play a tutorial video, while uncovering the lens to pause the video). When a learner watches a tutorial video on a smartphone, AttentiveLearner^2 uses both the front and back cameras of the smartphone as two complementary and fine-grained feedback channels: the back camera monitors her photoplethysmography (PPG) signals through fingertip transparency changes and the front camera tracks her facial expressions implicitly. AttentiveLearner^2 infers learners' affective and cognitive states during learning by analyzing their PPG signals and facial expressions in real-time. This paper offers three major contributions:

- Designing, prototyping, and evaluating a multimodal intelligent tutor to infer learners' cognitive and affective states in MOOCs on today's smartphones.
- A direct comparison of two modalities, i.e. the PPG channel and the facial expression analysis (FEA) channel, for predicting learners' emotional states and learning outcomes in the context of mobile MOOC learning.
- Proposing and evaluating a novel and effective feature set named Action Unit Variability (AUV) to capture the temporal dynamics of facial features.

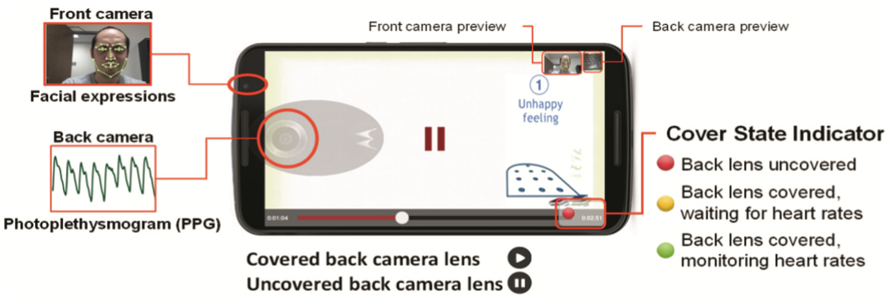


Fig. 1. The primary interface of *AttentiveLearner^2*, including two camera preview windows.

2 Related Work

Researchers have explored various approaches to understand and facilitate the consumption of educational videos in MOOCs. Server-side activity log, a.k.a. clickstream analysis, has been a primary information source to understand learners [6, 10, 19]. Guo and colleagues [6] discovered that shorter videos and Khan-style videos are more engaging. Kim et al. [10] further categorized different temporal video watching and pausing patterns for video playback. Van der Sluis and colleagues [19] reported that the watching time decreases when a video is either too difficult or too easy. Although activity logs are easy to collect and clickstream analysis can reveal insights in a scalable manner, behavior data from activity logs are usually sparse (e.g. one mouse click per video clip) and work better for *disclosing the aggregated trend* for revising the contents of courses in the future, rather than providing *personalized and adaptive support* for an individual learner.

Various techniques have been explored to engage MOOC learners, such as optimizing video production [10], real-time chat rooms [2], and social gamification [11]. Coetzee and colleagues [2] embedded a real-time chatroom supplementing an existing forum in a MOOC and found only 12% of their learners actively participated. Krause et al. [11] introduced social gamification elements to MOOCs, leading to a 25% increase in video watching time and a 23% increase in average scores. However, most of the proposed techniques require learners' active participation, e.g. joining discussions [2] or game activities [11]. In reality, most MOOC learners only watch lecture videos and skip optional activities [2]. As a result, it is still challenging to improve engagement and learning outcomes in MOOCs.

AttentiveLearner^2 is also relevant to existing research in affective computing [18]. Researchers have tried to model learners' affective and cognitive states [4, 5, 8, 13] automatically via physiological signals [8], facial expressions [5], or a combination of multiple modalities [4, 13]. For example, by combining features (i.e. feature fusion) from facial expressions, posture data, and dialog cues, D'Mello and Graesser [4] achieved approximately 0.2 improvements in Kappa for detecting 4 emotions in learning. Monkaresi et al. [13] ensembled heart rate and facial based models (model fusion) and improved the Area Under Curve (AUC) by approximately 0.1 when detecting engagement in essay writing. However, most existing approaches require dedicated sensors and PCs connected to high-speed Internet. Such requirements can prevent the wide adoption of affective technologies in real-world scenarios.

AttentiveLearner^2 builds on top of and extends AttentiveLearner [15, 16, 21, 22]. AttentiveLearner collects learners' PPG signals implicitly via the back camera during mobile MOOC learning, infers their affective and cognitive states [16, 21], and provides personalized interventions to improve learning outcomes [15, 22]. In comparison, AttentiveLearner^2 extends AttentiveLearner by adding a real-time facial expression channel via the front camera to gain a more robust emotion detection performance.

3 The Design of AttentiveLearner^2

3.1 On-Lens Video Control Interface

AttentiveLearner^2 uses on-lens finger gestures for tangible video control, i.e. a tutorial video is played when a learner covers and holds the back-camera lens and the video is paused when the back-camera lens is uncovered (Fig. 1). AttentiveLearner^2 extends the Static LensGesture algorithm [20] for lens-covering detection.

3.2 Dual-Camera Sensing System

AttentiveLearner^2 uses both the front and the back cameras of a smartphone as two complementary and fine-grained sensing channels. First, the back camera monitors a learner's PPG signals while she is watching a tutorial video. During learning, the arrival and withdrawal of fresh blood in every cardiac cycle change the learner's skin transparency, including her fingertip covering the back-camera lens. AttentiveLearner^2 employs the LivePulse algorithm [7] to extract normal to normal (NN) intervals from PPG signals. By detecting the peaks and valleys of these skin transparency changes (PPG signals), LivePulse infers the NN interval of heartbeats.

Second, the front camera tracks the learner's facial expressions in real-time. We use the Affdex SDK [12] to extract 30 facial values from each video frame. To improve learners' awareness of their facial alignment, AttentiveLearner^2 visualizes detected facial landmarks on the front camera preview widget whenever the learner's face is detected (Fig. 1). The facial preview window can be turned off by learners.

3.3 Emotion Detection

AttentiveLearner^2 infers learners' affective and cognitive states using machine learning models. The system can use PPG features, FEA features, or a combination of features from both channels (feature fusion).

PPG Features

We extract 8 dimensions of heart rate variability (HRV): (1) AVNN (average NN intervals); (2) SDNN (temporal standard deviations of NN intervals); (3) pNN60 (percentage of adjacent NN intervals with a difference longer than 60 ms); (4) rMSSD (root mean square of successive differences); (5) SDANN (standard deviation of the averages of NN interval within an m-second segment); (6) SDNNIDX (mean of the standard deviations of NN interval within an m-second segment); (7) SDNNIDX/rMSSD; (8) MAD (median absolute deviation). After discarding the first and the last 10 s of a video, we use a k-second non-overlapping sliding window (local) and the video window (global) to extract HRV features (Fig. 2). In total, we extract 16 features (PPG features) from each tutorial video.

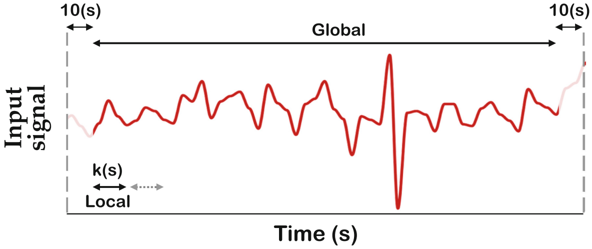


Fig. 2. PPG features and FEA features are extracted from each tutorial video.

FEA Features

It is worth noticing that when using facial expression features, most of today's systems extract and use action units (AUs) from short window frames (a few seconds) [4, 13]. While it is informative to identify the dominant facial expression at a specific moment, it may also be informative to understand the distribution, context, and dynamics of facial expressions during a longer learning process. In this study, we explore the feasibility of detecting aggregated emotion over one tutorial video.

Inspired by HRV features, we propose a new feature set, called Action Unit Variability (AUV), to capture the dynamics of facial features while a learner is watching a tutorial video. AUV has 8 dimensions: (1) AVAU (average action unit value); (2) SDAU (temporal standard deviations of action unit value); (3) MAXAU (the maximum value of action unit value); (4) rMSSD; (5) SDAAU (standard deviation of the averages of action unit value within an m-second segment); (6) SDAUIDX (mean of the standard deviations of action unit within an m-second segment); (7) SDAUIDX/rMSSD; (8) MAD. In each video, we extract 30 (Affdex outputs) × 8 (AUVs) × 2 (global/local window) = 480 features (FEA features) and select the top 16 features having the highest F-ratios from a univariate ANOVA test as in [4].

We intentionally replace pNN60 with a max pooling feature (MAXAU) in AUV. pNN60 is designed for NN intervals because it tracks the value changes every 60 units (milliseconds). Conversely, facial expressions do not change that frequently. For example, a learner would smile a few seconds creating a sudden peak in the signal during a 6-min video. Hence, a max pooling feature, monitoring signal peaks, is a better choice for FEA.

Feature Fusion

To balance the contribution of each modality, the feature fusion set has 16 features: the top 8 PPG features and the top 8 FEA features (selected by univariate ANOVA).

PPG features and FEA features have two temporal hyper-parameters: the sliding window size (k seconds) and the segment length (m seconds). We use grid search to optimize k in {60 s, 90 s, 120 s} and m in {3 s, 5 s, 10 s, 20 s, 30 s, 50 s, 60 s}. Features of each participant are normalized to zero mean and one standard deviation.

Prediction Models

We used SVMs with RBF-kernel to detect learners' affective and cognitive states. The models were trained and evaluated using leave-one-participant-out cross validation. Therefore, the reported results are from user-independent models. We performed parameter tuning for the gamma of RBF kernels, the tradeoff margins and the class-specific weights of SVMs.

4 Evaluation

4.1 Participant and Procedure

There were 29 participants (8 females) from a local university participating in our study. The average age was 25.2 ($\sigma = 4.5$). Following existing practices in handling outliers [3], we removed results from 3 participants because their self-reported ratings were almost identical across all experimental sessions. We used a within-subjects design in this study. Participants watched three 6-min tutorial videos (Fig. 3). The video topics were Astronomy (GammaRay), Learning Science (Learn2Learn), and Programming. The order of the video was randomized. After each video, participants took a quiz and reported 6 emotions (boredom, confusion, curiosity, frustration, happiness, and self-efficacy) during the video. The quiz contained 7 multiple choice questions and the emotional survey used 7-point Likert scale questions. Our experiment was conducted on a Nexus 6 smartphone.

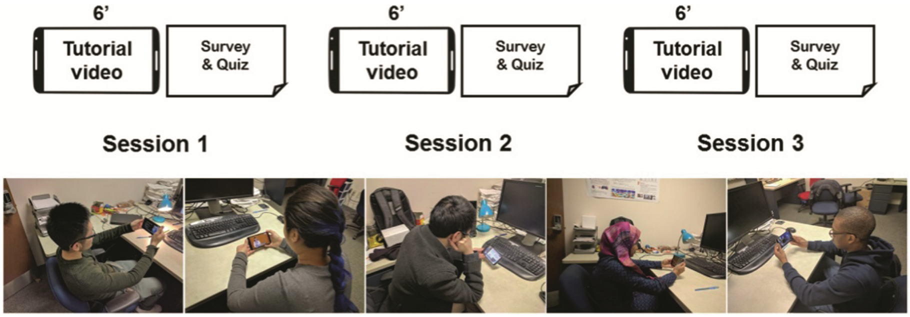


Fig. 3. The experimental procedure (top) and some participants in the experiment (bottom).

4.2 Results

Subjective Feedback

Overall, participants enjoyed using AttentiveLearner^2. Sample comments include: "*It's pretty easy to start using it*", "*The facial expression and pulse reader is cool*", and "*Keep attention of viewer*". At the same time, participants also raised some concerns about the battery life ("*[AttentiveLearner^2] drains battery more quickly*") and the facial preview widget ("*The picture from camera is distractive, especially when it changes because the face is not detected*" and "*sometimes the face monitor hid the slides out, which could be quite annoying*"). We conducted a battery stress test after the study and Attentive-Learner^2 can operate for 2 h and 2 min. Although the current battery life is not ideal, this duration is sufficient for mobile MOOC learning given the average learning time of a certificate earner in MOOC is 2–3 hours per week [21]. We are optimizing the battery life by adopting hardware decoding and reducing the sampling rate and sampling resolution of preview modes for both cameras.

Exploring Facial Features for Emotion Detection

We systemically explored the impact of different dimensions of AU features and facial emotion features (emotional features derived from the facial expression, such as anger, contempt, and disgust) on prediction performance. We are the first to explore most of the combinations via unmodified mobile devices. By building different detecting models, we investigated 20 dimensions of AU features and 10 dimensions of facial emotion features (FEA20 + 10) as well as two settings in previous studies, i.e. 5 dimensions of AU features (FEA5) [5] and 20 dimensions of AU features (FEA20) [4]. All models were trained with the same setting: using the top 16 AUV global and local features selected by univariate ANOVA. Figure 4 shows that including more AU features and facial emotion features can improve the system performance in 5 out of 6 emotions investigated. The only exception was Confusion where FEA20 (Kappa = 0.66) outperformed FEA20 + 10 (Kappa = 0.65).

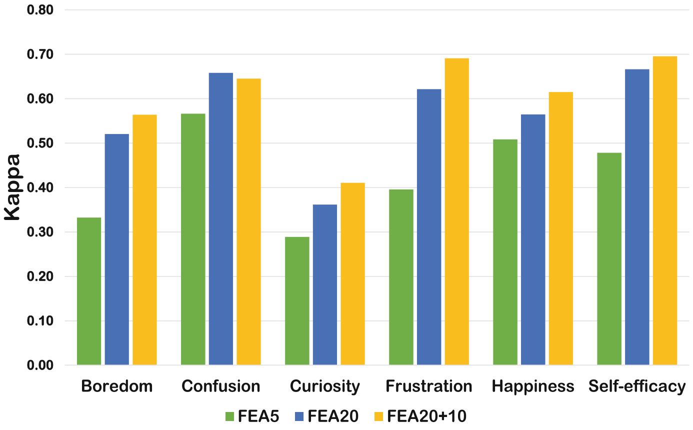


Fig. 4. The performance (Kappa) of three facial feature sets: FEA5 (5 AUs), FEA20 (20 AUs), and FEA20 + 10 (20 AUs + 10 high level emotions).

We found that AU1 (inner brow raise) and AU14 (dimpler) were the most discriminative features. In addition, two previously unexplored AUs contributed to the performance improvement of our FEA20 + 10 model, i.e. AU10 (upper lip raise) and AU18 (lip pucker).

We also found that certain AU and facial emotion features were not informative and can be skipped e.g. AU12 (lip stretch), AU9 (nose wrinkle), anger, and fear.

Emotion Detection Performance

Because of the unbalanced dataset, we reported Cohen's Kappa and AUC, in addition to the Accuracy metric. Jeni et al. [9] found Kappa and AUC are better than the Accuracy metric for skewed class labels. Table 1 shows the performance of detecting emotions via PPG-based models, FEA-based models, and models that combine these two modalities (feature fusion). All experimental models outperformed the majority vote baseline

Table 1. Performance on emotion detection.

Emotion	Majority	PPG			FEA			Feature fusion		
	Acc.	Acc.	Kap.	AUC	Acc.	Kap.	AUC	Acc.	Kap.	AUC
Boredom	70.5%	78.2%	0.35	0.67	84.6%	0.56	0.75	83.3%	0.57	0.86
Confusion	74.4%	78.2%	0.30	0.72	88.5%	0.65	0.71	84.6%	0.54	0.82
Curiosity	56.4%	74.4%	0.46	0.79	71.8%	0.41	0.72	73.1%	0.43	0.66
Frustration	78.2%	80.8%	0.22	0.46	91.0%	0.69	0.81	91.0%	0.71	0.82
Happiness	52.6%	70.5%	0.41	0.68	80.8%	0.61	0.82	80.8%	0.61	0.78
Efficacy	70.5%	79.5%	0.38	0.79	88.5%	0.70	0.82	87.2%	0.67	0.85
Average	67.1%	76.9%	0.35	0.67	84.2%	0.60	0.75	83.3%	0.59	0.86

(Majority). The best model (using feature fusion) achieved accuracy = 91.0%, Kappa = 0.71, and AUC = 0.82 while the worst model (using PPG features only) had accuracy = 80.8%, Kappa = 0.22, and AUC = 0.46 when detecting Frustration. The overall performance is very promising, considering that we did not use any additional sensors in this study. The FEA-based model had higher Kappa when detecting Boredom, Confusion, Frustration, Happiness, and Self-efficacy. The PPG-based model had higher Kappa when detecting Curiosity.

We found that PPG features and FEA features are complementary in detecting emotions. Combining PPG features and FEA features improved the Kappa of Boredom detection by 0.01 and Frustration detection by 0.02. The improvements of feature fusion models imply that both PPG features and FEA features are informative and complementary. Monkaresi et al. [13] also found an improvement when combining heart rate signals and facial expressions in predicting learners' engagement via dedicated sensors in desktop environments.

Emotion and Learning Outcome

We ran a regression analysis to evaluate the relations between learners' emotions and their learning outcomes. We used the probability outputs of our emotion detecting models as the input of the new regression model and its output is the quiz results. The emotions detected by PPG-based models, FEA-based models, and feature fusion models can explain approximately 20.6%, 50.6%, and 42.2% of the variability in the learning outcomes, respectively. The Boredom feature has a significant impact ($p < 0.01$ in the FEA-based and the feature fusion models) or a marginal impact ($p < 0.10$ in the PPG-based model) on the learning outcomes. Similarly, the Happiness feature in the FEA-based and feature fusion models has a significant impact ($p < 0.01$) on the learning outcome. Lastly, the Frustration feature detected by the FEA-based model has a significant impact ($p < 0.01$) on the learning outcomes. The results imply that a learner will not have a high learning outcome from a lesson if she feels bored or frustrated with the lesson.

5 Discussions and Future Work

This study shows the potential and advantages of using two complementary streams of physiological signals, i.e. PPG signals and facial expressions, to understand six emotions in learning (average accuracy = 84.4%). There are major efforts to translate higher prediction accuracy in learning to better learning outcomes. We plan to explore the use adaptive review approach by Pham and Wang [15] as an intervention technology in the near future. Instead of restricting the intervention to one review recommendation, we also plan to investigate the efficacy of recommending more than one review topics and recommending alternative activities such as quizzes.

It is worth noticing that the learning outcome predictions in Sect. 4.2 were made for each participant for each learning topic. Such predictions are hard to achieve with today's clickstream analysis techniques considering that there was only one finger tap for each video clip in our study. The prediction accuracy for learning outcome would be further improved if we use PPG features and FEA features, rather than emotions as input.

Moreover, since our dataset is imbalanced for many emotions, we plan to apply resampling techniques, such as down-sampling [4] or SMOTE [13], to further improve the robustness of our models.

The current studies were completed in a lab environment. We plan to conduct large-scale, longitudinal studies in learners' everyday environments in the near future. We shall make AttentiveLearner^2 freely available for public use and compatible with openEdX, a popular MOOC platform on the market. We also plan to explore visualization techniques to help instructors to identify difficulties among learners and opportunities for improvements in learning materials.

6 Conclusions

This paper reports an initial step towards a multimodal intelligent tutor named AttentiveLearner^2 for mobile MOOC learning on unmodified smartphones. The study shows the feasibility of capturing rich and fine-grained physiological signals such as PPG signals and facial expressions in mobile learning contexts without introducing any additional hardware. Experimental results show that PPG signals and facial expressions collected by AttentiveLearner^2 in real time are complementary and can serve as fine-grained, rich signals to understand learners' emotions. By capturing the temporal dynamics of both feature channels, AttentiveLearner^2 can achieve higher performance by combining both PPG features and FEA features. Our approach is complementary to today's existing technique such as clickstream analysis and is promising towards enabling personalized interventions for mobile MOOC learning.

References

1. Chuang, I., Ho, A.D.: HarvardX and MITx: four years of open online courses–Fall 2012-Summer 2016 (2016)
2. Coetzee, D., Fox, A., Hearst, M.A., Hartmann, B.: Chatrooms in MOOCs: all talk and no action. In: ACM Conference on Learning@ Scale, pp. 127–136. ACM (2014)
3. D'Mello, S.K., Dowell, N., Graesser, A.: Unimodal and multimodal human perception of naturalistic non-basic affective states during human-computer interactions. IEEE Trans. Affect. Comput. **4**(4), 452–465 (2013)
4. D'Mello, S.K., Graesser, A.: Multimodal semi-automated affect detection from conversational cues, gross body language, and facial features. User Model. User Adapt. Interact. **20**(2), 147–187 (2010)
5. Grafsgaard, J., Wiggins, J.B., Boyer, K.E., Wiebe, E.N., Lester, J.: Automatically recognizing facial expression: Predicting engagement and frustration. In: Educational Data Mining 2013 (2013)
6. Guo, P.J., Kim, J., Rubin, R.: How video production affects student engagement: An empirical study of MOOC videos. In: ACM Conference on Learning@ Scale, pp. 41–50. ACM (2014)
7. Han, T., Xiao, X., Shi, L., Canny, J., Wang, J.: Balancing accuracy and fun: designing engaging camera based mobile games for implicit heart rate monitoring. In: ACM Conference on Human Factors in Computing Systems, pp. 847–856. ACM (2015)

8. Hjortskov, N., Rissén, D., Blangsted, A.K., Fallentin, N., Lundberg, U., Søgaard, K.: The effect of mental stress on heart rate variability and blood pressure during computer work. Eur. J. Appl. Physiol. **92**(1–2), 84–89 (2004)
9. Jeni, L.A., Cohn, J.F., De La Torre, F.: Facing imbalanced data–recommendations for the use of performance metrics. In: Humaine Association Conference on Affective Computing and Intelligent Interaction, pp. 245–251. IEEE (2013)
10. Kim, J., Guo, P.J., Seaton, D.T., Mitros, P., Gajos, K.Z., Miller, R.C.: Understanding in-video dropouts and interaction peaks in online lecture videos. In: ACM Conference on Learning@ Scale, pp. 31–40. ACM (2014)
11. Krause, M., Mogalle, M., Pohl, H., Williams, J.J.: A playful game changer: Fostering student retention in online education with social gamification. In: ACM Conference on Learning@ Scale, pp. 95–102. ACM (2015)
12. McDuff, D., Mahmoud, A., Mavadati, M., Amr, M., Turcot, J., Kaliouby, R.e.: Affdex SDK: a cross-platform real-time multi-face expression recognition toolkit. In: ACM Conference on Human Factors in Computing Systems, pp. 3723–3726. ACM (2016)
13. Monkaresi, H., Bosch, N., Calvo, R.A., D'Mello, S.K.: Automated detection of engagement using video-based estimation of facial expressions and heart rate. IEEE Trans. Affect. Comput. **8**(1), 15–28 (2017)
14. Oviatt, S.: The Design of Future Educational Interfaces. Routledge, London (2013)
15. Pham, P., Wang, J.: Adaptive review for mobile MOOC learning via implicit physiological signal sensing. In: ACM International Conference on Multimodal Interaction, pp. 37–44. ACM (2016)
16. Pham, P., Wang, J.: AttentiveLearner: improving mobile MOOC learning via implicit heart rate tracking. In: Conati, C., Heffernan, N., Mitrovic, A., Verdejo, M. (eds.) AIED 2015. LNCS (LNAI), vol. 9112, pp. 367–376. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-19773-9_37
17. Pham, P., Wang, J.: AttentiveLearner^2: a multimodal approach for improving MOOC learning on mobile devices. In: André, E., Baker, R., Hu, X., Rodrigo, M., du Boulay, B. (eds.) AIED 2017. LNCS (LNAI), vol. 10331, pp. 561–564. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-61425-0_64
18. Pham, P., Wang, J.: Understanding emotional responses to mobile video advertisements via physiological signal sensing and facial expression analysis. In: The 22nd International Conference on Intelligent User Interfaces, pp. 67–78. ACM (2017)
19. Van der Sluis, F., Ginn, J., Van der Zee, T.: Explaining student behavior at scale: the influence of video complexity on student dwelling time. In: ACM Conference on Learning@ Scale, pp. 51–60. ACM (2016)
20. Xiao, X., Han, T., Wang, J.: LensGesture: augmenting mobile interactions with back-of-device finger gestures. In: ACM on International Conference on Multimodal Interaction, pp. 287–294. ACM (2013)
21. Xiao, X., Wang, J.: Towards attentive, bi-directional MOOC learning on mobile devices. In: ACM on International Conference on Multimodal Interaction, pp. 163–170. ACM (2015)
22. Xiao, X., Wang, J.: Understanding and detecting divided attention in mobile MOOC learning. In: ACM Conference on Human Factors in Computing Systems, pp. 2411–2415. ACM (2017)

Toward Tutoring Systems Inspired by Applied Behavioral Analysis

Michela Ponticorvo[1](✉), Angelo Rega[2], and Orazio Miglino[1]

[1] Department of Humanistic Studies, University of Naples "Federico II", Naples, Italy
michela.ponticorvo@unina.it
[2] IRFID, Institute for Research, Formation and Innovation on Disabilities,
Ottaviano, NA, Italy

Abstract. In this paper, we introduce an artificial tutoring systems inspired by Applied Behavioral Analysis, named ABA tutor.

Applied Behavioral Analysis is the application branch of analysis of behavior that derives from Behaviorism in psychology and has relevant features that can be transferred in an effective tutoring systems: the techniques of ABA are reproduced in the ABA tutor.

Moreover we describe the first implementation of ABA tutor and the application to olfactory learning, as a case-study. In more detail, the ABA tutor has been applied to SNIFF, an integrated software and hardware system conceived to train the sense of smell with a gamified approach, and has been tested on 84 people. Results indicate the effectiveness of ABA tutor in promoting olfactory learning, thus supporting that this tutor can be successfully introduced in learning environments.

Keywords: Applied Behavior Analysis
Intelligent Tutoring Systems · Olfactory learning

1 Introduction

In this paper, we propose to design and build an innovative family of tutoring systems that are based on Applied Behavioural Analysis (ABA). ABA is a methodology for psychological intervention, that has been developed starting in 1950s which translates in pragmatical and operational terms the scientific discoveries by behavioral psychologists. It represents the application branch of Analysis of Behavior, whose goal is to describe connection between behaviors displayed by living organisms and the variables that affect them [4].

ABA allows to support the acquisition of new behaviors by a person who is assisted by a professional operator (ABA therapist or behavioral technician).

Basically, ABA is a formalized and structured program to address and canalize learning processes towards adaptive outcomes; it is based on the rules of the law of effect, whose definition is apparently simple: an organism that is immersed in an environment, tends to repeat the behaviors that produce pleasant consequences whereas any behavior followed by unpleasant consequences is likely to be stopped [24,25].

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 160–169, 2018.
https://doi.org/10.1007/978-3-319-91464-0_16

The typical experimental setting conceived by Thorndike is the following: a cat is imprisoned in a cage, but it is free to explore the box to look for a lever that will allow to open the door and become free. The cat, even if it is in a constrained condition, is immersed in an environment offering many stimuli between which it must identify the stimulus that is useful to escape (the lever) and select from the behavioral repertory an action that will allow to open effectively the cage door.

Nowadays, thanks to the notable scientific effort that derived from this first formulation, lasting almost a century, a lot has been clarified about the law of effect. We now know both its neural, molecular bases [10] and learning and cognitive processes related to it: operant conditioning [21–23], latent learning [19], trial and error learning [1,24], just to cite.

Even if the law of effect cannot be considered, expecially for human beings, the main generating motor of learning [7,13] , it surely describes a basic condition to trigger cognitive processes with various complexity degrees: the organism, in function of its own sensorimotor features and of the constraints posed by the environment, decides autonomously which action to carry out and, if it will lead to positive effects, will tend to repeat that actions. It can be therefore applied to, at least, some human behaviors, namely the ones where associative processes are relevant.

Starting from this general framework, ABA has defined methodologies and techniques that allow to a professional operator to guide a learner to acquire some behaviors that have positive consequences so as to improve his/her psychological and social well-being. Indeed, Behavior Analysis theory and practice has given a relevant contribution to a rigorous reflection on learning processes [14].

In recent years, ABA has gained a privileged position in the treatment of children with Autistic Spectrum Disorder [20,26], allowing them to reduce problematic behaviors and establish a communicative language. It is also used in many others atypical development conditions. In the following section we introduce the ABA bases.

It is worth underlining that the ABA approach to tutoring systems is new, but it can be connected to the issue of adapting the frequency of prompts, which is a live vein in Intelligent Tutoring Systems research. For example, in a recent work by Bouchet et al. [2] it is investigated whether ITS using metacognitive strategies can be enhanced by variations in prompts based on learners' self-regulatory behaviors and the results indicate that the frequency of prompting has a relevant effect to promote learning, as confirmed by other studies [3,9,11]. The ABA approach, as it will be evident later, focuses on prompts, while keeping the attention on stimuli association and without considering metacognitive analysis or more complex behaviors. It is therefore fit to the most basic kinds of learning that are crucial in clinical condition such as Autistic Spectrum Disorder.

2 ABA Principles, Procedures and Techniques

The ABA technicians base their intervention of the following actions:

a. they plan and predispose the learning environment so as that the learner can identify the stimuli that are relevant for his/her with gradual ease;
b. they include the objects (or events) that, if manipulated or selected by the learner, produce some clear and evident consequences;
c. they arrange a reinforcement program, with, for example, the reward supply, so as that the learner, on acting in the learning environment, can evaluate, consciously or unconsciously, the consequences as neutral, positive or negative.

The planning of ABA intervention is based primarily on the definition of a learning environment where the target stimuli are clearly defined and are associated to precise behavioral responses by the learner that, if produced, are reinforced and rewarded. In the example of Thorndike experimental setting described above, the target stimulus is represented by the lever and the response to reinforce is the pressure that the cat must operate on the lever.

The main principles on which ABA is founded, in fact, are the ones of learning theory, law of effect and operant conditioning [12]. In the ABA framework, in fact, the behavior is seen as operant, as it produces effects and consequences in the environment and, in turn, it is affected by these effects. These principles are connected to the key concepts of Behaviorism: reinforce, stimuli, generalization [8].

These concepts are applied through four main procedures [18]. The first one, prompting, works as follows: some hints are provided, called indeed prompts, to the learner so as to facilitate the individuation of the target stimulus; for example, in the experimental condition reported above, the experimenter may guide the cat paw to press the lever that allows to exit the cage.

The second one, fading, is based on action that reinforce the responses also in presence of little changes of the target stimulus; for example, considering again the cage in Thorndike experimental procedure, the experimenter can include in the cage various levers that activate the exit mechanism also partially and reward with food the lever pressure by the cat.

The third one, shaping, is a procedure that is used to develop a behavior that doesn't belong to the usual behavior repertoire of an individual (or a species). As it is not possible to reinforce a behavior if it is not acted at least sometimes, the starting point is the reinforcement of a response that rarely appears and it is similar, also for some aspects only, to the desired behavioral response.

The last one, chaining, is a procedure that is useful to teach long behavioral sequences whose learning is easier if split in little behaviors. In the example, reported above, the target stimulus is represented by the lever and the response to reinforce is the pressure that the cat must operate on the lever. These principles have been integrated in the ABA tutor, that will be described in next section.

2.1 The ABA Tutor

The ABA tutoring systems includes two different modules and is illustrated in Fig. 1. The first module is the prompting module that can foresee and hint to the learner, anything that help to build the right association between stimuli and response. This module can be active or not, thus offering two options: with prompting or without prompting.

The second module is the stimuli exposure module which regulates the presentation of the different stimuli to the learner. This module can work in two ways: by randomly selecting stimuli or by following probabilistic rules.

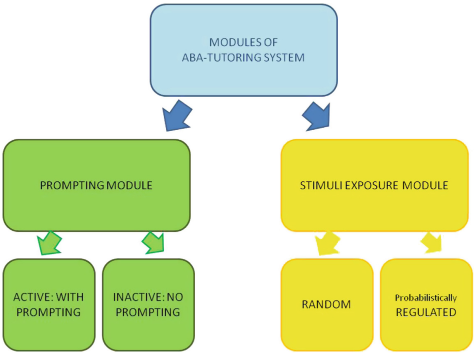


Fig. 1. The ABA tutoring system module

This probabilistic engine is further explained in Fig. 2. In this figure the stimuli are segregated on 4 categories to which a different and decreasing probability of exposure is associated. Category A includes stimuli that have never been presented or have never been recognized: the associated probability is 60%. In category B, there are the stimuli that have been recognized at least once and have a probability to be presented again of 25%. Category C hosts stimuli that have been recognized at least 2 times and are associated to the probability of 10%. The last category includes stimuli that have been recognized at least 3 times and have the probability of 5% to be presented again.

At the beginning of a learning cycle all stimuli are in Category A, then the stimuli begin to follow a flow, represented by the arrow in the figure that moves them to the other categories. The percentage associated with each category is fixed, but the stimulus can change category from A to D. In principle, at the end, all stimuli should be in Category D.

In the following subsection the tool where the ABA tutor is implemented is described.

3 SNIFF: An Integrated System to Develop the Sense of Smell

In humans, olfaction, a chemical sense, is believed to be a secondary way to know the world around, but it plays an important role for cognitive functions.

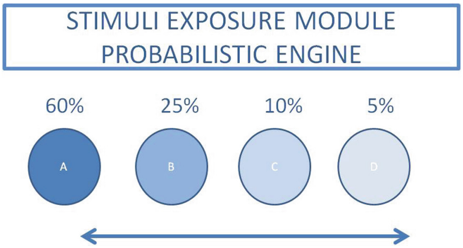


Fig. 2. The probabilistic engine of the stimuli exposure module

Olfaction can be fruitfully exploited to produce learning environments, becoming the starting point for multisensory applications which put together digital tools and physical materials, thus fostering motivation and engagement by users, also with special needs [5] This sense has some features that give it remarkable peculiarities for learning, as the strong connection between olfactory stimuli and emotions. In fact, on a neural level, this connection is due to the anatomy of the olfactory pathways that are directly routed to the cortex, without passing through the thalamus. Odour information is transmitted directly to the limbic system, associated with memory and emotional processes [6]. As an effect, it can influence mood, acquisition of new information, and use of information in many different contexts including social interactions, thus affecting learning.

Moreover, it is difficult that olfactory learning may benefit from techniques that exploit semantics, as it is typically associative learning. A tool that can be both used to assess and train the sense of smell is SNIFF. SNIFF is an application which is intended to assess and stimulate the sense of smell. It is represented in Fig. 3.

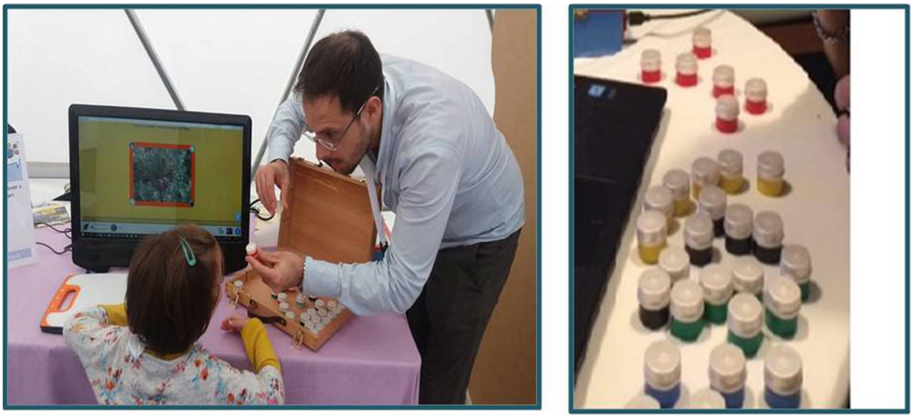


Fig. 3. On the left, Sniff system with the PC, the table and the jars. On the right, the smelling jars with the color prompt (Color figure online)

SNIFF derives directly from the Montessori pedagogical approach [16], being a technology-enhanced version of the materials named smelling bottles or jars, which are included in the sensorial area of Montessori classroom. SNIFF is a game with 50 attempts, each consisting in the association of a smell with the corresponding image that is proposed by the software. These exercises reside in the database, which collects the smell activities. This activity helps to refine the olfactory sense in children and in adults. The materials used in SNIFF augments 30 smelling jars with RFID that can be read by the table, where the antenna resides. The smelling jars thus become hybrid educational materials with a physical and digital side, which enhance the traditional smelling bottles. SNIFF is implemented using STELT (Smart Technologies to Enhance Learning and Teaching) [15], an integrated software and hardware platform to build educational materials. It allows to design and implement learning materials based on artificial intelligence and tangible interfaces, for example physical objects which are equipped with RFID sensors, as tools to support user-computer interactions [17].

STELT includes three modules: the first one is devoted to storyboarding and allows to build customized personalized scenarios; the second one is based on recording that offers the functionalities to rack users' data interaction; the third one implements adaptive tutoring, that delivers on-time feedbacks.

3.1 The ABA Tutor in SNIFF

The tutoring system in SNIFF is the ABA tutor with the two module described above.

Prompting is implemented as follows: in order to facilitate recognition, jars are segregated into 5 groups of 6 odours, identified by a coloured jar (the colour is assigned by chance). The system asks for the selection of a specific smell, within a reduced group of jars, identified with the chromatic categorization (i.e. the request is search only between the blue and black jars). This way, the learner can exploit the prompts provided by the jars colors (Fig. 3).

The stimuli exposure module is implemented through the modification of the game in response to the learner's selection. In particular, if the player fails to recognize a certain stimulus, the associated probability to be proposed increases, otherwise, if the odour is correctly recognized, it decreases. This procedure works, adapting to the player abilities in two different ways:

- If the player fails the recognition, SNIFF decreases the difficulty level reducing the numerosity of the groups to look for the odour within; if the player is able to recognize the stimulus, in the next exercise the level will become higher, widening the group.
- If the player fails the recognition of a certain odour, the associated probability to be proposed increases, otherwise, if the odour is correctly recognized, it decreases.

Let us give an example of how the SNIFF tutoring system works. The participants starts from an intermediate level where he/she is asked to recognize a certain smell between two sub-groups ("Find the honey smell. It is in a yellow or blue jar"). If he/she fails, next time he/she will be asked to find it in a smaller groups ("It is in a blue jar"); if he/she correctly identifies the smell, he/she moves to a more difficult level ("Find the orange smell. It is in a yellow, blue or green jar"). In case of failure, the probability of presentation of a certain smell increases. SNIFF gives an appropriate and immediate feedback to the player. If he/she finds the correct smell, the system shows on the screen a little anecdote on the found smell. If the choice is incorrect, it says which was the right odour or invites to go on searching.

SNIFF and the ABA tutor have been tested on more than 80 people, in the experiment that is described below, in order to verify the effectiveness of the probabilistic engine and prompts.

4 Materials and Method

The experiment is meant to test the 4 conditions of ABA tutor that derive from the combination of the two modules. In other words from the combination of the two modules with two conditions each, we define 4 experimental conditions. More in detail, the 4 different conditions are: TUTOR selection (T) and RANDOM selection (R), crossed with PROMPTS (P) and WITHOUT PROMPTS (nP). This experimental design was conceived in order to understand if there are differences in the performance of olfactory recognition due to tutoring functions (probabilistic engine) or to the facilitation provided by visual cues (prompts). These cues are the colors on the jars.

4.1 Participants and Procedure

Participants are 84 people, 42 males and 42 females, 52 non-smokers (62% of the sample) and 32 smokers (38%). The average age is 30.61. None of the participants had any neurological impairment. Participants are equally distributed and randomly assigned to one of the four experimental conditions.

The experimental procedure is the following: the experimenter introduces the participant to the experimental session, telling that it is a game on olfactory abilities. During each session, the SNIFF software presents on the screen some images, for example a fruit or a plant and the participant has to look for the jar whose smell corresponds to the image. For example, if SNIFF presents a cookie, the participant has to find the cookie smelling jar. The experimenter does not help participants to accomplish the task in anyway. When all sessions are over, participants are briefly interviewed to have feedbacks about the tool.

5 Results

The participants recognize an average of 20.27 stimuli on 30 attempts (st.dev. 4.89) ranging from a minimum of 8 to a maximum of 29. Considering the correct response on 50 attempts (stimuli are presented more than once), the average recognition is 29.53 (st.dev. 7.03), minimum 14, maximum 46 (Fig. 4).

There is no significant correlation between age and stimuli recognition ($r = -0.145$; $p = 0.19$) and age and overall performance ($r = -0.123$; $p = 0.263$). The experimental condition in which the participants can rely on the adaptive tutor and the colour prompts T-P leads to perform better on stimuli recognition (on 30 stimuli av. = 24.61, st.dev. = 2.95; on 50 attempts av. = 32.88, st.dev. = 4.84). The T-nP, with the adaptive tutor but no colour prompt generates a lower recognition rate (on 30 stimuli av. = 20.77, st.dev. = 4.12; on 50 attempts av. = 26.54, st.dev. = 5.89).

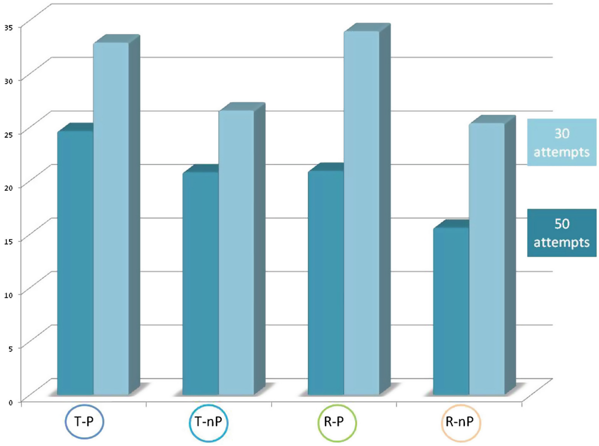


Fig. 4. Average recognized stimuli in the four experimental conditions

Considering the 2 conditions with the random selection, the colour prompt gives relevant advantage: R-P on 30 stimuli av. = 20.90, st.dev. = 3.22; on 50 attempts av. = 33.95, st.dev. = 5.71, giving the best overall performance; whereas R-nP on 30 stimuli av. = 15.59, st.dev. = 4.52; on 50 attempts av. = 25.36, st.dev. = 7.16.

The role of the ABA tutor is also investigated with a One-way ANOVA with tutor function as independent variable and stimuli recognition as dependant one. Results indicate a significant effect of the tutor: $F(1, 82) = 19.272$; $p < 0.01$. Moreover, the colour prompt has significant effects both on stimuli recognition $F(1, 82) = 20.93$; $p < 0.01$ and overall performance $F(1, 82) = 33.279$; $p < 0.01$.

These data indicate that the ABA tutor, with both components, is effective in increasing smelling performance.

6 Conclusions and Future Directions

ABA has obtained a great success in many different contexts, in modifying human behavior: in the case of children with Autism Spectrum disorder, it gives them the chance to build a behavioral repertoire assuring a better quality of life at a personal level and giving a confirmation of the effectiveness of this technique and its procedure to impact on people learning processes on a scientific level.

It is nonetheless worth underlining that the ABA is not limited to the mere application of this technique, but it is a comprehensive procedure/methodology that has a maieutical aspect. Who adopts this procedure has to promote the emergence and the stabilization of one or more behaviours, helping to child to show it, to act it. As Socrates helped to find the personal thoughts in each person, the ABA technicians helps to find the own behavior inside each child. In this view, the ABA technician cannot be replaced by the ABA artificial tutor, but the ABA-tutor can enter this process as a valid complement for the technician work. With the help of ABA tutor, the ABA expert can focus on the maieutical process while the ABA-tutor deals with technical aspects, recording and tracing the interaction, offering personal profiles, highlighting strengths or weaknesses.

The ABA tutor can be moreover applied to a variety of contexts, in particular, on one side, all the contexts where it is important to reinforce active behavior, in opposition with the traditional TEL where respondents behaviors are taken into account more frequently and, on the other, all the situations where associative learning is relevant. The tutoring system inspired by ABA can be further developed by adding new functions related to other ABA procedures and further tested with different learning materials and at different ages.

References

1. Boswell, F.P.: Trial and error learning. Psychol. Rev. **54**(5), 282 (1947)
2. Bouchet, F., Harley, J.M., Azevedo, R.: Can adaptive pedagogical agents' prompting strategies improve students' learning and self-regulation? In: Micarelli, A., Stamper, J., Panourgia, K. (eds.) ITS 2016. LNCS, vol. 9684, pp. 368–374. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39583-8_43
3. Bouchet, F., Harley, J.M., Azevedo, R.: Impact of different pedagogical agents' adaptive self-regulated prompting strategies on learning with metatutor. In: Lane, H.C., Yacef, K., Mostow, J., Pavlik, P. (eds.) AIED 2013. LNCS (LNAI), vol. 7926, pp. 815–819. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39112-5_120
4. Cooper, J.O., Heron, T.E., Heward, W.L.: Applied Behavior Analysis. Prentice Hall, Upper Saddle River (2007)
5. Di Fuccio, R., Ponticorvo, M., Ferrara, F., Miglino, O.: Digital and multisensory storytelling: narration with smell, taste and touch. In: Verbert, K., Sharples, M., Klobučar, T. (eds.) EC-TEL 2016. LNCS, vol. 9891, pp. 509–512. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-45153-4_51
6. Firestein, S.: How the olfactory system makes sense of scents. Nature **413**(6852), 211 (2001)
7. Gamez, A.M., Rosas, J.M.: Associations in human instrumental conditioning. Learn. Motiv. **38**, 242–261 (2007)
8. Granpeesheh, D., Tarbox, J., Dixon, D.R.: Applied behavior analytic interventions for children with autism: a description and review of treatment research. Ann. Clin. Psychiatr. **21**(3), 162–173 (2009)

9. Harley, J.M., Taub, M., Azevedo, R., Bouchet, F.: "Let's set up some subgoals": understanding human-pedagogical agent collaborations and their implications for learning and prompt and feedback compliance. IEEE Trans. Learn. Technol. **11**, 54–66 (2017)
10. Kandel, E.R., Klein, M., Castellucci, V.F., Schacher, S., Goelet, P.: Some principles emerging from the study of short-and long-term memory. Neurosci. Res. **3**(6), 498–520 (1986)
11. Kinnebrew, J.S., Gauch, B.C., Segedy, J.R., Biswas, G.: Studying student use of self-regulated learning tools in an open-ended learning environment. In: Conati, C., Heffernan, N., Mitrovic, A., Verdejo, M.F. (eds.) AIED 2015. LNCS (LNAI), vol. 9112, pp. 185–194. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-19773-9_19
12. Martin, G., Pear, J.: Strategie e tecniche per il cambiamento. La via comportamentale. McGraw-Hill Education, New York (2000)
13. Gamez, A.M., Rosas, J.M.: Transfer of stimulus control across instrumental responses are attenuated by extinction in human instrumental conditioning. Int. J. Psychol. Psychol. Therapy **5**, 3 (2005)
14. Mazur, J.E.: Learning and Behavior. Routledge, Abingdon (2015)
15. Miglino, O., Di Ferdinando, A., Di Fuccio, R., Rega, A., Ricci, C.: Bridging digital and physical educational games using RFID/NFC technologies. J. e-Learn. Knowl. Soc. **10**(3), 87–104 (2014)
16. Montessori, M.: The Montessori Method. Transaction Publishers, London (2013)
17. Ponticorvo, M., Di Fuccio, R., Di Ferdinando, A., Miglino, O.: An agent-based modelling approach to build up educational digital games for kindergarten and primary schools. Expert Syst. **34**(4) (2017)
18. Ricci, C., Romeo, A., Bellifemine, D., Carradori, G., Magaudda C.: Il Manuale ABA-VB. Applied Behavior Analysis and Verbal Behavior. Fondamenti tecniche e programmi di intervento, Erickson (2014)
19. Seward, J.P.: An experimental analysis of latent learning. J. Exp. Psychol. **39**(2), 177 (1949)
20. Shook, G.L.: An examination of the integrity and future of the Behavior Analyst Certification Board credentials. Behav. Modif. **29**, 562–574 (2005)
21. Skinner, B.F.: Operant behavior. Am. Psychol. **18**(8), 503 (1963)
22. Skinner, B.F.: The Behavior of Organisms. Appleton-Century-Crofts, New York (1938)
23. Skinner, B.F.: Science and Human Behavior. Free Press, New York (1953)
24. Thorndike, E.L.: A proof of the law of effect. Science **77**, 173–175 (1933)
25. Thorndike, E.L.: The law of effect. Am. J. Psychol. **39**(1/4), 212–222 (1927)
26. Virués-Ortega, J.: Applied behavior analytic intervention for autism in early childhood: meta-analysis, meta-regression and dose-response meta-analysis of multiple outcomes. Clin. Psychol. Rev. **30**, 387–399 (2010)

The Role of Negative Emotions and Emotion Regulation on Self-Regulated Learning with MetaTutor

Megan J. Price(✉), Nicholas V. Mudrick, Michelle Taub, and Roger Azevedo

North Carolina State University, Raleigh, USA
{mjprice3,nvmudric,mtaub,razeved}@ncsu.edu

Abstract. Self-regulated learning (SRL) and emotion regulation have been studied as separate constructs which impact students' learning with intelligent tutoring systems (ITSs). There is a general assumption that students who are proficient in enacting cognitive and metacognitive SRL processes during learning with ITSs are also proficient emotion regulators. In this paper, we investigated the relationship between metacognitive and cognitive SRL processes and emotion regulation by examining students' self-perceived emotion regulation strategies and comparing the differences between their (1) mean self-reported negative emotions, (2) proportional learning gains (PLGs), and the frequency of (3) metacognitive and (4) cognitive strategy use as they interacted with MetaTutor, an ITS designed to teach students about the circulatory system. Students were classified into groups based on self-perceived emotion regulation strategies and results showed students who perceived themselves as using adaptive emotion regulation strategies reported less negative emotions. Although no significant differences were found between the groups' learning outcomes, there were significant differences between the groups' frequency use of cognitive and metacognitive processes throughout the task. Our results emphasize the need to better understand how real-time emotion regulation strategies relate to SRL processes during learning with ITSs and can be used to enhance learning outcomes by encouraging adaptive emotion regulation strategies as well as increased frequencies of metacognitive and cognitive SRL processes.

Keywords: Emotion regulation · Self-regulated learning · Metacognition
Emotions · Intelligent Tutoring Systems

1 Introduction

Self-regulated learning (SRL) increases students' understanding of various topics and domains [1] and includes the accurate monitoring and regulation of their cognitive, affective, metacognitive, and motivational (CAMM) processes [2]. Research shows higher frequencies of cognitive strategies and metacognitive processes within Intelligent Tutoring Systems (ITSs) have been shown to contribute to higher learning outcomes and are typically indicative of competent self-regulators [2, 3]. While some research primarily focuses on the cognitive and metacognitive aspects of SRL, affective

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 170–179, 2018.
https://doi.org/10.1007/978-3-319-91464-0_17

processes, such as emotion regulation, have been studied as separate constructs outside of SRL and ITS communities. As such, this study addresses this gap by investigating the impact of students' self-perceived emotion regulation strategies on their self-reported emotions, learning outcomes, and their frequency use of cognitive and metacognitive strategies during learning with MetaTutor.

Emotion regulation is an example of an affective process whereby an individual seeks to influence their emotions by either down-regulating their negative emotions (e.g., confusion, anger, frustration), up-regulating their positive emotions (e.g., joy, curiosity) or suppressing the expression of their emotions entirely [4]. Research regarding emotions during learning with ITSs has shown negative emotions like confusion, frustration, and boredom to be harmful to learning goals if experienced for a significant time threshold [5, 6]. According to Gross's Process Model of Emotion Regulation, there are two primary strategies of emotion regulation: cognitive reappraisal (CR) and expressive suppression (ES) [4]. CR is considered to be an adaptive emotion regulation strategy as the amount of negative emotions a student experiences can be down-regulated *before* they become detrimental to the student's learning. Conversely, ES is considered a maladaptive emotion regulation strategy because the negative emotions are suppressed, therefore potentially disrupting the student's learning. Despite extensive research in different areas of psychology, there is little research in the area of computing and ITS on the impact of emotion regulation during learning with ITSs [7].

Previous studies have found instructing individuals to engage in CR to be successful in helping individuals achieve higher learning outcomes and increasing student engagement [8]. However, few studies have sought to understand a student's emotion regulation as it relates to their metacognitive and cognitive SRL processes during learning. Despite this, there is the assumption that students who are proficient self-regulators are equally capable emotion regulators. This may be due to research regarding increased frequencies of SRL processes leading to higher learning outcomes and the assumption that accurate monitoring and regulation of learning translates to accurate monitoring and regulation of emotions. While it is likely that proficient emotion regulators are also proficient self-regulators, this relationship has not yet been firmly established, especially during learning with ITSs.

This study represents an initial examination of the role of self-reported emotion regulation strategies, negative emotions, and their relationship to cognitive and metacognitive SRL processes during learning with MetaTutor, a hypermedia-based ITS. For this study we analyzed students' metacognitive and cognitive SRL processes with log-file data and their self-reported emotion regulation strategies and emotions throughout their learning session in an effort to understand the relationship between SRL and emotion regulation during the students' interaction with MetaTutor. Our research questions are as follows:

1. Are there significant differences between conditions and emotion regulation groups' self-reported negative emotions as reported on the EV?
2. Are there significant differences between conditions and emotion regulation groups on proportional learning gain (PLG)?
3. Are there significant differences in the frequency of metacognitive process use between emotion regulation groups?

4. Are there significant differences in the frequency of cognitive strategy use between emotion regulation groups?

2 Methods

2.1 Participants

One hundred three (n = 103) college students (55% female) participated in this study. The participants' ages ranged from 18–35 (M = 20.31, SD = 2.42) and they were compensated $10 per hour for their participation for completing the two-day experiment.

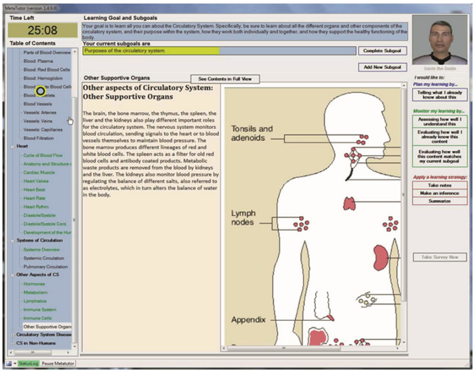
2.2 MetaTutor: A Hypermedia-Based ITS for Biology

MetaTutor is an ITS designed to teach students about the circulatory system, through 47 pages of text and static images, with the aid of four pedagogical agents (PAs), with only one PA visible to the student at any given point in time [2]. Each PA is responsible for a specific SRL process or strategy during the learning session as students should engage in SRL while learning about the human circulatory system. More specifically, Mary the Monitor monitors and prompts the use of metacognitive processes such as judgments of learning (JOLs), feeling of knowing (FOK), and content evaluation (CE) of the relevancy of the material given the current subgoal. Sam the Strategizer prompts students with cognitive strategies such as taking notes (TN), or summarizing what they have just read (SUMM), and provides feedback on the accuracy of their summaries during learning. Pam the Planner helps students activate their prior knowledge about the content (PKA) as well as plan sub-goals (e.g. *Purpose of the Circulatory System*, *Path of Blood Flow*) at the beginning of the learning session based on the students' text input in the interaction log with Pam. Gavin the Guide guides the students through the environment and administers self-report questionnaires such as the Emotions and Values Questionnaire [9] throughout the session and the Emotion Regulation Questionnaire [10] during the pretest. The interface also includes a timer, overall learning goal, subgoals set by the student with Pam, the PA, SRL palette, text and diagrams, and table of contents to foster students' monitoring (see Fig. 1).

2.3 Materials and Equipment

During the 90-min learning session multichannel process, product, and self-report data were collected including: videos of facial expressions of emotion, electrodermal activity, eye-tracking data, log-file data, a biology pretest and posttest, and several self-report measures. These self-report measures include questionnaires regarding students' basic and learning-centered emotions, epistemic beliefs, and motivation.

The two self-report measures used in this study include the Emotions and Values Questionnaire (EV; [9]) and the Emotion Regulation Questionnaire (ERQ; [10]). The EV consists of 20 questions and asks students about their emotions and values on a Likert scale ranging from 1 to 5 based on Pekrun et al.'s [11] model of academic achievement



Your goal is to learn all you can about the Circulatory System. Specifically, be sure to learn about all the different organs and other components of the circulatory system, and their purpose within the system, how they work both individually and together, and how they support the healthy functioning of the body.

Your current subgoals are

Purposes of the circulatory system

Other aspects of Circulatory System: Other Supportive Organs

The brain, the bone marrow, the thymus, the spleen, the liver and the kidneys also play different important roles for the circulatory system. The nervous system monitors blood circulation, sending signals to the heart or to blood vessels themselves to maintain blood pressure. The bone marrow produces different lineages of red and white blood cells. The spleen acts as a filter for old red blood cells and antibody coated products. Metabolic waste products are removed from the blood by kidneys and the liver. The kidneys also monitor blood pressure by regulating the balance of different salts, also referred to as electrolytes, which in turn alters the balance of water in the body.

Fig. 1. Screenshot of the MetaTutor interface.

emotions. The EV was administered by Gavin approximately every 14 min to ensure multiple samples from each student. The questionnaire asked the students how they were currently feeling and contained "Right now I feel:" followed by either a positive, negative, and neutral emotion. The positive emotions included: pride, joy/happiness, hope, curiosity/interest, eureka/sudden understanding, and surprise (observed range: 5–24). The negative emotions included: anger/frustration, fear, contempt/disgust, sadness, confusion, hopelessness and boredom (observed range: 8–32).

The ERQ was designed to measure students' tendency to regulate their emotions through either CR (range: 6–42, α = .79) or ES (range: 4–28, α = .73) via a 10-item, 7-point Likert-type scale. CR is defined as the capacity to control attention to emotional stimuli and to change the mental representation of that stimuli, whereas ES is the capacity to hide emotions from being perceived by others in a social setting [10]. According to Gross and John [10], each subscale of the ERQ is independent of the other, meaning a person who rates themselves as more likely to use CR as an emotion regulation strategy is no more or less likely to use ES as an emotion regulation strategy.

2.4 Procedure

Students were randomly assigned to either a *control* or a *prompt and feedback* condition before participating. In the *control* condition, no prompts or feedback were given during the learning session. In the *prompt and feedback* condition, MetaTutor's PAs provide real-time adaptive scaffolding. Students were prompted by the PAs to engage in cognitive and metacognitive SRL processes and were then provided with feedback on their performance based on a combination of time- and activity-based production rules, as well as experimental condition. For example, a time-based production rule for the PKA cognitive strategy occurs when the student has started a new subgoal. Additionally, if a summary is less than three sentences, Sam will prompt the students to revise their summary, which is an example of an activity-based production rule. In both conditions, students could willingly engage in SRL strategies without being prompted by clicking on the SRL palette.

The experimental session took place over the course of two days. On Day 1 students were administered the informed consent, demographics questionnaire, biology pretest, and several self-report measures of emotion. The Day 1 session took approximately 30–60 min. The biology pretest was comprised of 30 multiple-choice questions related to the circulatory system, which was used to assess prior knowledge. Day 2 began with equipment calibration, followed by introductory videos about how to navigate through MetaTutor and the various SRL strategies, a sub-goal setting phase with Pam, and then the 90-min learning session. Students also had the opportunity to take and revise notes, take quizzes, and other embedded assessments such as the EV and a revised Agent Persona Inventory [12].

Following the learning session, students were administered the posttest, which was equivalent to the pretest and developed by a subject matter expert, and self-report questionnaires (i.e., EV, a revised Agent Persona Inventory, Achievement Emotions Questionnaire [13]), debriefed, paid, and thanked for their time.

2.5 Data Sources and Coding of Learning Outcomes and Self-Report Measures

Although multichannel data were collected, for this study only log-file and self-report questionnaire data were used. Log-file data included the frequency and duration of metacognitive or cognitive SRL strategy use that were either prompted by a PA or initiated by the student. Cognitive strategies included TN, SUMM, INF, and PKA. Metacognitive processes included JOL, FOK, and CE. Only the student-initiated actions were considered for this study to include students from both conditions and to study the frequency of self-initiated attempts at regulating their learning compared to their self-reported emotion regulation strategies.

Overall learning about biology with MetaTutor was assessed by comparing pretest and posttest scores using proportional learning gain (PLG) using Witherspoon et al.'s [14] formula.

Additionally, this study examined the ERQ, and responses were coded such that each student received two scores, one for each subscale with the observed range for cognitive reappraisal as 7–42 and 5–23 for expressive suppression. To determine a student's primary emotion regulation strategy, a median split was conducted on the subscale scores such that those who fell *at* or *below* the median received a "low" rating for that particular strategy, and those who fell above the median received a "high" rating (CR *Mdn.* = 31, ES *Mdn.* = 15). This 2 × 2 structure created four possible groups: High CR-High ES, High CR-Low ES, Low CR-Low ES, Low CR-High ES.

Furthermore, students completed multiple instances of the EV [9] throughout their learning session. For the purpose of this study, the emotions were grouped into positive or negative emotions in order to create a composite score. A sample question for a positive emotion would be "Right now I feel that I am enjoying myself", and "Right now I feel anger" for a negative emotion. Students completed the questionnaire multiple times throughout their learning session, at approximately 14 m intervals. However, students could choose to postpone the questionnaire, which resulted in different frequencies of EV administrations per student (*M* EV administration = 6.84; range of EV administration was 1–10). To account for this, an average score across the entire session was calculated per student such that each student had one positive emotion (e.g., joy, pride, hope) mean score and one negative emotion (e.g., confusion, frustration, boredom) mean score. For this study, only negative emotions were examined due to the evidence from multiple studies that negative emotions have a greater impact on a student's learning [5, 10, 15].

3 Results

3.1 Research Question 1: Are There Significant Differences Between Conditions and Emotion Regulation Groups' Self-Reported Negative Emotions as Reported on the EV?

A 2 × 4 factorial ANOVA using emotion regulation groups and condition as the independent variables revealed a significant main effect on students' mean self-reported negative emotions reported on the EV ($F(3, 95) = 2.75$, $p = .047$, $\eta_p^2 = .08$, Table 1). The main effect for condition was not significant ($F(1, 95) = 0.51$, $p = .475$) and the interaction effect for emotion regulation group and condition was also not significant ($F(3, 95) = 0.06$, $p = .982$). Tukey's HSD post-hoc comparisons revealed that students in the High CR-Low ES group reported significantly less negative emotions (i.e., lower scores in response to experiencing feelings of confusion, frustration, anxiety, etc.) than students in the Low CR-Low ES group. No significant differences were found between the other emotion regulation groups' self-reported negative emotions.

Table 1. Means and standard deviations and post-hoc comparisons of negative emotions between emotion regulation groups

	n	*M(SD)*	Tukey's HSD comparisons			
			L. CR-L. ES	L. CR-H. ES	H. CR- L. ES	H. CR-H. ES
L. CR-L. ES	30	16.55 (4.76)	-	−0.05	3.04*	1.05
L. CR-H. ES	21	16.60 (3.74)	.05	-	3.09	1.09
H. CR-L. ES	25	13.51 (3.69)	−3.04*	−3.09	-	−1.99
H. CR-H. ES	27	15.50 (4.27)	−1.05	−1.09	1.99	-

*p <.05.
Note. CR = Cognitive reappraisal, ES = Expressive suppression, H. = High, L. = Low.

3.2 Research Question 2: Are There Significant Differences Between Conditions and Emotion Regulation Groups on Proportional Learning Gain (PLG)?

A 2 × 4 factorial ANOVA revealed no significant main effects across conditions ($F(1, 95) = 0.52$, $p = .474$) or emotion regulation groups on PLGs ($F(3, 95) = 1.52$, $p = .215$, Table 2). Additionally, there was no significant interaction effect for emotion regulation groups and condition ($F(3, 95) = 0.64$, $p = .592$).

Table 2. Means and standard deviations of proportional learning gains between emotion regulation groups.

	n	*M (SD)*
Low CR-Low ES	30	.17 (.07)
Low CR-High ES	21	.26 (.08)
High CR-Low ES	25	.37 (.07)
High CR-High ES	27	.23 (.07)

Note. CR = Cognitive reappraisal, ES = Expressive suppression.

3.3 Research Question 3: Are There Significant Differences in the Frequency of Metacognitive Process Use Between Emotion Regulation Groups?

Frequencies of student-initiated instances were totaled across three different possible metacognitive processes on the SRL palette (JOL, CE, and FOK) per student. A 3 × 4 chi-square test revealed significant differences between frequency of metacognitive process use across ERQ groups (χ^2 (6) = 28.21, $p < .001$, Table 3). Overall, the results indicate that JOLs were the most frequently used metacognitive processes and CEs were the least frequently used by all students regardless of emotion regulation group. In addition, students in the Low-ES groups used more metacognitive processes than those in the High-ES groups.

Table 3. Chi-square results of metacognitive processes by emotion regulation groups.

	H. CR-H. ES	H. CR-L. ES	L. CR-H. ES	L. CR-L. ES	χ^2	Total
JOL freq.	116	208	117	209	28.21*	650
FOK freq.	51	61	58	60		230
CE freq.	11	49	23	65		148
Total	178	318	198	334		

*$p < .001$.
Note. JOL = Judgment of Learning, FOK = Feeling of Knowing, CE = Content Evaluation, H. = High, L. = Low.

3.4 Research Question 4: Are There Significant Differences in the Frequency of Cognitive Strategy Use Between Emotion Regulation Groups?

Frequencies of student-initiated instances were totaled across four cognitive strategies (i.e., PKA, SUMM, INF, and TN) per student. A 4 × 4 chi-square test revealed significant differences between frequency of cognitive strategy use across ERQ groups (χ^2 (9) = 85.06, $p < .001$, Table 4). Overall, the results indicate that both Low-ES groups used more cognitive strategies than the High-ES groups. In addition, all groups took more notes than any other learning strategy.

Table 4. Chi-square results of cognitive strategies by emotion regulation groups.

	H. CR-H. ES	H. CR-L. ES	L. CR-H. ES	L. CR-L. ES	χ^2	Total
PKA freq.	11	8	5	11	85.06*	35
SUMM freq.	20	84	18	23		145
INF freq.	10	12	11	14		47
TN freq.	220	246	194	407		1067
Total	261	350	228	455		

*$p < .001$.
Note. PKA = Prior Knowledge Activation, SUMM = Summary, INF = Inference, TN = Take Notes, H. = High, L. = Low.

4 Discussion and Future Directions

Understanding the relationship between emotion regulation and SRL processes is key to enhancing future ITSs and the development of intelligent PAs. The purpose of this study was to examine the role of students' self-perceived emotion regulation strategies and mean negative emotions and their relationship with students' learning outcomes and the frequency of their metacognitive processes and cognitive strategies during learning with MetaTutor. More specifically, our results indicated students in the High CR-Low ES group reported significantly less negative emotions during their learning session compared to the other three emotion regulation groups. However, no significant differences were found between these emotion regulation groups and their proportional learning gains. Furthermore, students in the High CR-Low ES and Low CR-Low ES groups used more metacognitive processes and cognitive strategies.

The results of our first research question support existing literature regarding the benefits of using cognitive reappraisal as an adaptive emotion regulation strategy due to the High CR-Low ES group self-reporting significantly lower mean negative emotions than the Low CR-Low ES group. Being able to automatically identify students' emotional states and track the changes as they occur in real time can allow researchers to detect adaptive emotion regulation strategies such as CR. For example, if an ITS is able to detect a student's confusion through facial detection software and observes the level of confusion decrease, this could be indicative of CR. The student's facial expressions of emotions can then be fed to the ITS, and the PA can in turn praise the student's use of an adaptive emotion regulation strategy and encourage them to continue using CR. These results strongly suggest the advantages of a PA who can recognize CR when it is occurring and encourage students to continue to engage in such a process.

While the High CR-Low ES group reported less negative emotions than the Low CR-Low ES group, there were no significant differences in learning outcomes. Additionally, all groups enacted more JOLs than any other metacognitive process, suggesting future iterations of ITSs do not need to prompt students to engage in JOLs. Instead, the focus should be placed on prompting students to engage in CEs and FOKs as students did not enact these processes as often as JOLs although they are just as beneficial to overall learning. Similarly, students engaged in TNs more than the other cognitive strategies, again suggesting future iterations of PAs do not need to prompt this strategy, but instead focus on prompting students to use PKAs, SUMMs, and INFs because they are also beneficial to a student's learning. Furthermore, both the High CR-Low ES and Low CR-Low ES groups self-initiated a higher frequency of cognitive and metacognitive SRL processes. The high frequency of SRL processes by the Low CR-Low ES group contradicts the underlying assumption that proficient self-regulators are proficient emotion regulators because we would expect that students who perceive themselves to not engage in emotion regulation strategies would not frequently engage in SRL processes because we would expect their emotion regulation strategies to translate to their metacognitive and cognitive SRL strategies. This suggests that more in-depth analyses of the quality of the processes and strategies used are necessary to understand the differences between these groups and why the differences in frequencies are not aligned with their self-perceived emotion regulation strategies. For example, it is possible that students engaged in more strategies, but the quality of those strategies could have been higher and were not as effective as a student who engaged in less, but more effective strategies.

Future research should investigate the qualitative differences of the SRL processes deployed by students and seek to determine how to identify adaptive emotion regulation strategies in a more empirical method than self-report measures. For example, using empirical methods to detect emotion regulation as it occurs in real time by using facial recognition software can provide researchers with a more accurate representation of a student's emotion regulation strategies instead of relying on the student's self-perceived emotion regulation strategies. Combining this empirical data with a more in-depth analysis of the student's metacognitive and cognitive SRL strategy use affords researchers a more comprehensive understanding of the relationship between SRL processes and emotion regulation as they occur during learning with ITSs which can then be used to develop adaptive and intelligent PAs that scaffold students' SRL and emotion regulation strategies and provide encouragement and feedback to the students.

Acknowledgments. This research was supported by funding from the National Science Foundation (DRL#1431552; DRL#1660878, DRL#1661202) and the Social Sciences and Humanities Research Council of Canada (SSHRC 895-2011-1006). Any opinions, findings, conclusions, or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation or Social Sciences and Humanities Research Council of Canada. The authors would like to thank the members from the SMART Lab at NCSU for their assistance with data collection.

References

1. Schunk, D.H., Greene, J.A. (eds.): Handbook of Self-Regulation of Learning and Performance, 2nd edn. Routledge, New York (2018)
2. Azevedo, R., Taub, M., Mudrick, N.V.: Understanding and reasoning about real-time cognitive, affective, and metacognitive processes to foster self-regulation with advanced learning technologies. In: Schunk, D.H., Greene, J.A. (eds.) Handbook of Self-Regulation of Learning and Performance, 2nd edn, pp. 254–270. Routledge, New York (2018)

3. Winne, P.H.: Cognition and metacognition within self-regulated learning. In: Schunk, D.H., Greene, J.A. (eds.) Handbook of Self-Regulation of Learning and Performance, 2nd edn, pp. 36–48. Routledge, New York (2018)
4. Gross, J.J.: Emotion regulation: current status and future prospects. Psychol. Inq. **26**, 1–26 (2015)
5. D'Mello, S., Graesser, A.: Dynamics of affective states during complex learning. Learn. Instr. **22**, 145–157 (2012)
6. D'Mello, S.K., Craig, S.D., Sullins, J., Graesser, A.C.: Predicting affective states through an emote-aloud procedure from AutoTutor's mixed-initiative dialogue. Int. J. Artif. Intell. Educ. **16**, 3–28 (2006)
7. Bosse, T., Pontier, M., Treur, J.: A computational model based on Gross' emotion regulation theory. Cogn. Syst. Res. **11**, 211–230 (2010)
8. Strain, A.C., D'Mello, S.K.: Affect regulation during learning: the enhancing effect of cognitive reappraisal. Appl. Cogn. Psychol. **29**, 1–19 (2015)
9. Harley, J.M., Bouchet, F., Hussain, M.S., Azevedo, R., Calvo, R.: A multi-componential analysis of emotions during complex learning with an intelligent multi-agent system. Comput. Hum. Behav. **48**, 615–625 (2015)
10. Gross, J.J., John, O.P.: Individual differences in two emotion regulation processes: implications for affect, relationships, and well-being. J. Pers. Soc. Psychol. **85**, 348 (2003)
11. Pekrun, R., Elliot, A.J., Maier, M.A.: Achievement goals and discrete achievement emotions: a theoretical model and prospective test. J. Educ. Psychol. **98**, 583 (2006)
12. Baylor, A., Ryu, J.: The API (Agent Persona Instrument) for assessing pedagogical agent persona. In: EdMedia: World Conference on Educational Media and Technology, pp. 448–451. Association for the Advancement of Computing in Education (AACE) (2003)
13. Pekrun, R., Götz, T., Perry, R.P.: Achievement emotions questionnaire (AEQ). User's manual. Department of Psychology, University of Munich, Munich, Germany (2005)
14. Witherspoon, A.M., Azevedo, R., D'Mello, S.: The dynamics of self-regulatory processes within self-and externally regulated learning episodes during complex science learning with hypermedia. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) ITS 2008. LNCS, vol. 5091, pp. 260–269. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-69132-7_30
15. Pekrun, R., Goetz, T., Frenzel, A., Barchfeld, P., Pery, R.: Measuring emotions in students' learning and performance: the Achievement Emotions Questionnaire (AEQ). Contemp. Educ. Psychol. **36**, 36–48 (2011)

Analysis of Permanence Time in Emotional States: A Case Study Using Educational Software

Helena Reis[1](✉), Danilo Alvares[2](✉), Patricia Jaques[3](✉), and Seiji Isotani[1](✉)

[1] Instituto de Ciências de Computação e Matematica Computacional (ICMC),
Universidade de São Paulo (USP), São Carlos, SP, Brazil
{helenamcd,sisotani}@icmc.usp.br

[2] Harvard T.H. Chan School of Public Health, Boston, MA, USA
dalvares@hsph.harvard.edu

[3] Programa de Pos-Graduação em Computação Aplicada (PPGCA),
Universidade do Vale do Rio dos Sinos (UNISINOS), São Leopoldo, RS, Brazil
pjaques@unisinos.br

Abstract. This article presents the results of an experiment in which we investigated how prior algebra knowledge and personality can influence the permanence time from the confusion state to frustration/boredom state in a computer learning environment. Our experimental results indicate that people with a neurotic personality and a low level of algebra knowledge can deal with confusion for less time and can easily feel frustrated/bored when there is no intervention. Our analysis also suggest that people with an extroversion personality and a low level of algebra knowledge are able to control confusion for longer, leading to later interventions. These findings support that it is possible to detect emotions in a less invasive way and without the need of physiological sensors or complex algorithms. Furthermore, obtained median times can be incorporated into computational regulation models (e.g. adaptive interfaces) to regulate students' emotion during the teaching-learning process.

1 Introduction

Emotions have an important impact on learning, accelerating or hindering it [1–3]. Although most studies that investigate emotions in the educational context have focused on the basic emotions (e.g. sadness, anger, joy, or surprise), recent research provides evidence that non-basic emotions (e.g. engagement/flow, confusion, frustration, and boredom) are more frequent in computer-based learning (CBL) [4].

Contrary to common sense, confusion is an emotion that should not be avoided in the learning context because it makes the students seek the knowledge and maintain focus and attention and is related to encouragement [3]. Students commonly experience confusion in complex activities which occur throughout the entire school period. When confusion is detected and experienced, students need to engage in cognitive activities to solve their confusion.

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 180–190, 2018.
https://doi.org/10.1007/978-3-319-91464-0_18

Although confusion has been positively correlated with learning [5,6], it should be regulated according to students' personality and prior knowledge to have an adequate duration [6]. If confusion persists for a long time, it can become frustration or boredom [7]. For instance, if a student has a neurotic personality (i.e., tends to have negative emotions) and he/she is a beginner in the subject or the task is complex, the confusion must be managed cautiously so as not to become boredom. However, one question remains: how specifically do students' personality and prior knowledge affect the permanence time in a confusion state?

To verify the relation of the permanence time in a confusion state with the students' personality and their prior knowledge on learning problems, we have developed an experiment with higher education students from three Brazilian public universities. This experiment was performed in more than 70 h and involved 30 randomly selected students, 2 instructors and 2 coders, who analyzed the prior algebra knowledge and the students' personality. These students were also asked to solve algebra problems in a computer learning environment. The results obtained can be used to create less invasive and low cost alternative emotion detectors, unlike other types of detection, such as physiological sensors, which can be costly and make students uncomfortable [8]. In addition, this approach supports emotional regulation models, in which the students' emotion can be regulated when they are feeling some negative emotion.

2 Related Works

Emotion is a state constantly awakened and lived by individuals [9]. Emotions can undergo several changes upon receiving a stimulus (i.e., a person can become angry, sad, or joyful). This change of emotion is called the transition state and can be influenced by initial emotion, emotional events, prior knowledge and individual personality characteristics [10].

Studies [11,12] have showed that, in CBL, emotions transit between engagement/flow, confusion and frustration/boredom. Confusion and engagement/flow have been positively correlated with learning, while frustration and boredom have been negatively correlated. Ideally, emotions should be regulated considering students' personality and knowledge, and should have a certain duration [6]. For instance, academic risk theory contrasts adventurous students, who want to be challenged with difficult tasks, take the risk of failure, and manage negative emotions when it occurs, with students who take less risks, avoiding complex tasks and effectively minimize learning situations in which they are likely to fail and experience negative emotions [13–15].

In addition, it is necessary to identify who could benefit from an inductive intervention for a particular emotion [5,6]. Of course, confusing a beginner student or inducing confusion during high-risk learning activities is not a sensible strategy. Nowadays, these interventions are ideally suited for gifted students who get bored and lose interest in activities without challenges [12,16].

[17] investigated the emotional transitions[1], during the teaching-learning process (Table 1). Their results show evidence that confusion can lead to two other emotions: engagement/flow and frustration.

Table 1. Expected affective transitions (adapted from [17]).

Time t_i	Time $t_i + 1$			
	Boredom	Flow	Confusion	Frustration
Boredom		-	-	?
Flow	-		+	-
Confusion	-	+		+
Frustration	+	-	?	

(+) indicates that the transaction is expected.
(-) indicates that the transaction is highly unlikely.
(?) indicates that there is no explicit relation in the model.

When confusion is not handled appropriately (i.e., when instructors do not monitor the duration of confusion or the students' personality - tendency to emotional states) the student can become frustrated and then bored. When a student experiences negative emotions, such as frustration and boredom, he/she tends to remain in this state and not transit to positive emotional states, such as engagement/flow. On the other hand, students in flow state tend to remain engaged or transit alternately to confusion, which is positively correlated with learning [6]. In a more general context, the transition from an emotional state to another one can be modeled through a survival or reliability analysis, where the transition probabilities are obtained for each specified time [19].

3 Method

During the teaching-learning process, the student can experience several emotions (engagement/flow, confusion, frustration/boredom, etc.) and these emotions can transit from one to another. The change from an emotional state to another one can depend on several factors, including personality and prior knowledge on the subject. For instance, a beginner student with personality tending to negative emotions (e.g., neuroticism) can easily move from confusion to frustration/boredom when he/she strives to solve a problem or has a long period of confusion.

This section describes the design and planning used in our experiment to investigate whether the personality traits (neuroticism and extroversion) and

[1] As students can feel more than one emotion each time, in this paper we are considering the dominant emotion in a moment in time and the transition to another dominant emotion [18] during the teaching-learning process.

algebra knowledge of students affect their permanence times from the confusion state to frustration/boredom state during the use of a computer learning environment.

3.1 Research Questions

This work aims to answer the following research questions (RQ):

RQ$_1$: Do personality traits influence the permanence time from confusion to frustration/boredom in an educational software?

RQ$_2$: Does algebra proficiency influence the permanence time from confusion to frustration/boredom in an educational software?

RQ$_3$: What is the average permanence time spent by a student (with different combinations of personality traits and algebra proficiency) from confusion to frustration/boredom in an educational software?

3.2 Participants

We gathered information from 30 randomly selected students. 13% are women (corresponding to 4 people) aged 21–22 years (mean age of 21.5 years) and 87% are men (corresponding to 26 people) aged 19–34 years (mean age of 26.5 years). All participants were invited through direct contact and are undergraduate students in areas related to Computer Science and Software Engineering, except 4 male participants who are undergraduate students in Industrial Design (2 people), Production Engineering (1 person) and Geography (1 person).

3.3 Materials

For the execution of the experiment, the following instruments were used: (i) questionnaire with personal questions, (ii) multiple-choice test, (iii) personality trait scale and (iv) equation solving test. The personal questionnaire aims to know the profile of participants and contains questions about personal data, such as whether the participants have already used some educational software before and their prior knowledge on algebra. The algebra test covered five multiple choice questions which involved first and second degree equations, determinants, factorials, and logarithms. These questions were suggested by two math teachers and they were separated into three difficult levels: basic (2), intermediate (2) and difficult (1). Each correct question was assigned the value 0.2 points, so the total points for each student can be 0, 0.2, 0.4, 0.6, 0.8 or 1.

The personality trait scale[2] assessed the participant's personality for neuroticism and extroversion indexes. Each of these indexes varies from 0 to 1, where 1 indicates greater presence of characteristic summarized by the index. This test was based on the five-factor model and is written in Portuguese. Finally, nine algebra questions were proposed in an educational software[3], all in the same

[2] Available at https://personalitatem.ufs.br/inventory/home.xhtml.

[3] Available at http://acubo.tecnologia.ws/aluno.html.

scope previously tested. The nine questions also involved first and second degree equations, determinants, factorials, and logarithms. They were suggested by two math teachers and they were separated into three difficult levels: basic (3), intermediate (4) and difficult (2).

3.4 Procedure

Each participant had an hour and a half to perform the experiment (Fig. 1). First, students were asked to fill in the personal questionnaire in 10 min. Afterwards, students answered the multiple choice test and then the personality trait scale, which had a total duration of 30 min. After this initial phase, students accessed a system for solving algebra problems. First, they should fill out information with their personal data. Hence, they solved nine algebra problems (scratch papers were provided) and entered their final answer in the system. The students' face were recorded while they were solving the equations for later analyses of their emotions (confusion, frustration and boredom).

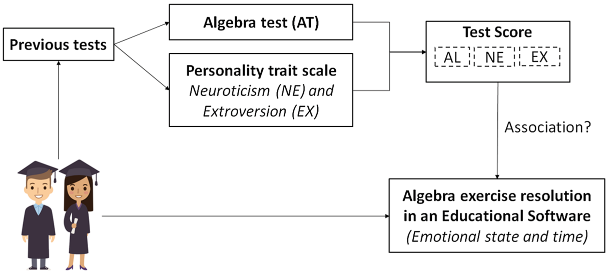

Previous tests
Algebra test (AT)
Personality trait scale
Neuroticism (NE) and Extroversion (EX)
Test Score
AL
NE
EX
Association?
Algebra exercise resolution in an Educational Software
(Emotional state and time)

Fig. 1. Procedure of the experiment.

As previously mentioned, the analysis of the video was performed by two coders, one graduates from Computer Science and one from Industrial Design. According to [20], the recognition of facial expressions by humans has an accuracy of approximately 87%, making it possible that people with no training in Psychology and without any tool to measure emotions can recognize different types of emotions by the face.

The coders annotated the emotions students experienced and the permanence time in each of them during problem solving. They separately annotated the beginning and ending time that each student expressed, by face, the emotions engagement/flow, confusion, frustration, or boredom. Then, they discussed together the annotations and reached an agreement. The coding of the facial expressions was performed according to guidelines suggested by [21], in which they propose 10 heuristics of human behavior in order to infer what emotions humans are experiencing at a given moment.

4 Statistical Modeling

This study aims to determine the permanence time from a confusion state to a frustration/boredom state for each student. So we used a statistical model that describes the permanence time in a state until an event of interest occurs. This type of approach is known as survival or reliability analysis, and its main objective is to know the behavior of a given population as to the time of occurrence of one or more events of interest [22]. We have opted for a Bayesian inferential analysis, in which all unknown quantities (e.g., model parameters) can be modeled by means of probability distributions [23].

Our modeling for the permanence time from confusion to frustration/boredom for i^{th} student, $i = 1, \ldots, 30$, will be characterized by a Weibull proportional hazards model with fragility [24], given by:

$$h_i(t \mid \boldsymbol{\theta}, w_i, \mathbf{x}_i) = \lambda \, \alpha \, t^{\alpha-1} \exp\Big(\beta_1 \, x_{1i} + \beta_2 \, x_{2i} + \beta_3 \, x_{3i}\Big) \, w_i, \tag{1}$$

where $h_i(t \mid \cdot)$ is a risk function for student i at time t. The parameters β_1, β_2 and β_3 are the coefficients associated with students scores i in the preliminary algebra test (x_{1i}), neuroticism index (x_{2i}) and extroversion index (x_{3i}), respectively. λ and α are scale and shape parameters of the Weibull distribution that defines the baseline risk function described by $\lambda \, \alpha \, t^{\alpha-1}$. The frailty (or random effect) for the student i is described by $w_i \sim \text{Gamma}(\eta, \eta)$, where its variance is given by $\kappa = 1/\eta$. The parameter and variable vectors are defined as $\boldsymbol{\theta} = (\beta_1, \beta_2, \beta_3, \lambda, \alpha, \eta)^\top$ and $\mathbf{x}_i = (x_{1i}, x_{2i}, x_{3i})^\top$.

We assume prior independence as a default specification. In addition, we elicit vague proper marginal prior distributions, in order to give all inferential prominence to the data:

$$\begin{gathered} \pi(\beta_1) = \pi(\beta_2) = \pi(\beta_3) = \pi(\log(\lambda)) = \text{Normal}(0, 1000), \\ \pi(\alpha) = \pi(\eta) = \text{Gamma}(0.01, 0.01). \end{gathered} \tag{2}$$

The posterior distribution $\pi(\boldsymbol{\theta} \mid \mathcal{D})$ is not obtained analytically ($\mathcal{D}$ represents the collected data), so we approximate it using Markov chain Monte Carlo (MCMC) [25] with the `WinBUGS` software [26].

5 Discussion of Results

In this paper, we aim to study the permanence time of students from confusion to frustration/boredom, given their algebra knowledge and personality. The presented results below are from the model (1) with the marginal prior distributions defined as in (2), where we use the following MCMC configuration: 3 Markov chains with 500000 iterations (after burning of 50000) and storage every 500 iterations to reduce autocorrelation in the posterior sample.

Table 2 summarizes the posterior estimates of the parameters β_1, β_2 and β_3 of the model (1) with prior distributions (2).

Table 2. Posterior summary of the parameters of interest for the time from confusion to frustration/boredom.

Parameter	Mean	SD	2.5%	50%	97.5%
β_1	−1.970	0.672	−3.334	−1.956	−0.725
β_2	0.721	0.634	−0.602	0.725	1.943
β_3	−0.828	0.710	−2.202	−0.835	0.581

The interpretation of the results from the Bayesian approach is simple and fundamentally based on quantities of interest, such as mean, standard deviation (SD) and quartiles, from probability distributions, called posterior (or *a posteriori*) distributions. In addition, the interpretation of mean signal of each parameter is counter-intuitive, because in the case of negative signal, the higher the value of the variable referring to this parameter, the longer the time until the student experiences the event of interest.

As for the question $\mathbf{RQ}_2$, based on presented results in Table 2, we have evidences that the more algebra knowledge, the longer the time until the student in the confusion state becomes frustrated/bored with the exercise, i.e., there is a positive association. The answer to $\mathbf{RQ}_1$ is divided into two parts, where the first one refers to extroversion index and the second one to neuroticism index. The interpretation for extroversion index is analogous to the algebra knowledge, since the higher the extroversion index, the longer the permanence time between confusion and frustration/boredom. On the other hand, an increase in the neuroticism index leads to a reduction in the permanence time from confusion to frustration/boredom, i.e., the student gives up on the problem solving more quickly (negative association).

From the posterior distribution of $\boldsymbol{\theta}$, we can calculate derived quantities that help us to answer $\mathbf{RQ}_3$, such as the median time of permanence time from confusion to frustration/boredom. This median time T is obtained when the survival function $S_i(T)$ for a student i takes the value 0.5 and is given by:

$$T = \left[\frac{-\log(0.5)}{\lambda \exp\Big(\beta_1\, x_{1i} + \beta_2\, x_{2i} + \beta_3\, x_{3i}\Big)\, w_i}\right]^{1/\alpha}. \tag{3}$$

It is worth mentioning that, in survival analysis, right-censored data make the median more informative than the mean, i.e., participants who do not experience the event of interest - frustration/boredom state - during the study time would take the average to higher values. In Bayesian terms, we can calculate the posterior mean of the median time (3) for a generic individual i, given his/her variables $\mathbf{x}_i$, by the following equation:

$$
\begin{aligned}
\mathrm{E}\big[S_i(T \mid \boldsymbol{\theta}, \mathbf{x}_i) \mid \mathcal{D}\big] &= \int S_i(T \mid \boldsymbol{\theta}, w_i, \mathbf{x}_i)\, \pi(w_i, \boldsymbol{\theta} \mid \mathcal{D})\, \mathrm{d}(w_i, \boldsymbol{\theta}) \\
&\approx \frac{1}{K} \sum_{k=1}^{K} S_i(T \mid \boldsymbol{\theta}^{(k)}, w_i^{(k)}, \mathbf{x}_i),
\end{aligned} \tag{4}
$$

where $\boldsymbol{\theta}^{(k)}$ and $w_i^{(k)}$ are k^{th} values of posterior sample $\pi(w_i, \boldsymbol{\theta} \mid \mathcal{D})$. The approximation (4) is carried out by Monte Carlo integration [27].

To exemplify the obtained results and answer $\mathbf{RQ}_3$, Table 3 shows the posterior mean of median time of the permanence time from confusion to frustration/boredom (4) with different configurations of prior algebra knowledge and scores of neuroticism and extroversion indexes.

Table 3. Posterior mean of median time of the permanence time from confusion to frustration/boredom for different profiles.

Variable	Profile								
x_1	0	0.5	1	0	0	1	1	0	1
x_2	0	0.5	0	1	0	1	0	1	1
x_3	0	0.5	0	0	1	0	1	1	1
Time	34	76	159	19	64	90	313	37	175

From the results, the influence of each variable at median time of the permanence time from confusion to frustration/boredom is evident. For instance, a student with a "median" configuration (i.e., 0.5 for all variables) would take on average 76 s to migrate between the states of interest. We also can note that students with a maximum score (value 1) for the prior algebra test and the extroversion index, and a minimum score (value 0) for the neuroticism index would require, on average, 313 s to pass from confusion to frustration/boredom. On the other hand, when the student has a maximum score for the neuroticism index and a minimum for the prior algebra test and the extroversion index, he/she takes, on average, 19 s. Note that we did not include in the model (1) a covariate describing the difficulty level of algebra questions. This is due to the fact that, in our preliminary analyzes, this covariate did not present relevant differences between the three difficult levels (basic, intermediate and difficult).

6 Threats to Validity

A possible threat to validity of the results is the representativeness of the sample, since all individuals who participated in the study are undergraduate students. In this way, it is not possible to generalize the results to the entire student population. From the statistics point of view, this problem can be circumvented with repetitions of this study in different samples of undergraduate students.

Although there was concern in assessing the permanence time from confusion to frustration/boredom, the usability of the software for the algebra test may have prevented some students to complete the exercises. Another threat to be considered is the use of two people to code students emotions by face observation, leading to an interpersonal bias as to the accuracy in the permanence time.

7 Conclusions

We aimed to study the permanence time of students from a confusion state to a frustration/boredom state, given their algebra knowledge and personality. The results of our experiment suggest that prior algebra knowledge and personality traits affect the permanence time from confusion to frustration/boredom during the learning process. Notably, students with a high neuroticism index and low score in the algebra test cannot deal very well with the confusion, remaining less time in this emotion compared to students with high extroversion index and low score in the algebra test. This means that neurotic students who are also beginners in algebra spend less time confused and feel frustration/boredom more quickly compared to extroverted student with the same level of algebra knowledge.

We believe that these preliminary results can help in the elaboration of computational models of emotional regulation of students. The permanence time in the confusion state can be integrated with other information, for instance, physiological sensors or automatic facial expressions. This information can contribute to emotion control using interfaces, which detect personality traits and the beginning of the confusion state, adapting elements for beginner students with little tolerance in the permanence time of confusion. In addition, as future work, the results may help with the investigation of student self-efficacy. Other benefit of this experiment was the provision of a replication package[4], which can be used by other researchers for the same purpose.

Focusing on the statistical approach, the Bayesian perspective for survival analysis was of paramount importance, since we had a small data set and wanted to interpret derived quantities based on model parameters (questions $\mathbf{RQ}_1$, $\mathbf{RQ}_2$ and $\mathbf{RQ}_3$). Furthermore, the estimated permanence time between confusion and frustration/boredom for any new student profile is easily calculated, providing quick decision-making with respect to emotional regulation.

References

1. Pekrun, R.: The control-value theory of achievement emotions: assumptions, corollaries, and implications for educational research and practice. Educ. Psychol. Rev. **18**(4), 315–341 (2006)
2. Sullins, J., Graesser, A.C.: The relationship between cognitive disequilibrium, emotions and individual differences on student question generation. Int. J. Learn. Technol. **9**(3), 221–247 (2014)

[4] http://goo.gl/YtGn7H.

3. D'Mello, S., Graesser, A.: AutoTutor and affective autotutor: learning by talking with cognitively and emotionally intelligent computers that talk back. ACM Trans. Interact. Intell. Syst. **2**(4), 23:1–23:39 (2013)
4. D'Mello, S., Calvo, R.A.: Beyond the basic emotions: what should affective computing compute? In: CHI 2013 Extended Abstracts on Human Factors in Computing Systems, pp. 2287–2294 (2013)
5. Craig, S., Graesser, A., Sullins, J., Gholson, B.: Affect and learning: an exploratory look into the role of affect in learning with AutoTutor. J. Educ. Media **29**(3), 241–250 (2004)
6. D'Mello, S., Picard, R.W., Graesser, A.: Toward an affect-sensitive AutoTutor. IEEE Intell. Syst. **22**(4), 53–61 (2007)
7. Graesser, A., D'Mello, S.K.: Theoretical perspectives on affect and deep learning. In: Calvo, R., D'Mello, S. (eds.) New Perspectives on Affect and Learning Technologies, vol. 3, pp. 11–21. Springer, New York (2011). https://doi.org/10.1007/978-1-4419-9625-1_2
8. Shanabrook, D.H., Arroyo, I., Woolf, B.P.: Using touch as a predictor of effort: what the iPad can tell us about user affective state. In: Masthoff, J., Mobasher, B., Desmarais, M.C., Nkambou, R. (eds.) UMAP 2012. LNCS, vol. 7379, pp. 322–327. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-31454-4_29
9. Xiaolan, P., Lun, X., Xin, L., Zhiliang, W.: Emotional state transition model based on stimulus and personality characteristics. China Commun. **10**(6), 146–155 (2013)
10. Gross, J.J.: Emotion regulation: affective, cognitive, and social consequences. Psychophysiology **39**(3), 281–291 (2002)
11. Graesser, A.C., Chipman, P., Haynes, B.C., Olney, A.: AutoTutor: an intelligent tutoring system with mixed-initiative dialogue. IEEE Trans. Educ. **48**(4), 612–618 (2005)
12. D'Mello, S., Lehman, B., Pekrun, R., Graesser, A.: Confusion can be beneficial for learning. Learn. Instr. **29**, 153–170 (2014)
13. Clifford, M.M.: Failure tolerance and academic risk-taking in ten- to twelve-year-old students. Br. J. Educ. Psychol. **58**(1), 15–27 (1988)
14. Dweck, C.S.: Mindset: The New Psychology of Success, 1st edn. Random House Incorporated, New York (2006)
15. Meyer, D.K., Turner, J.C.: Re-conceptualizing emotion and motivation to learn in classroom contexts. Educ. Psychol. Rev. **18**(4), 377–390 (2006)
16. Pekrun, R., Götz, T., Daniels, L.M., Stupnisky, R.H., Perry, R.P.: Boredom in achievement settings: exploring control-value antecedents and performance outcomes of a neglected emotion. J. Educ. Psychol. **102**(3), 531–549 (2010)
17. D'Mello, S.: Monitoring affective trajectories during complex learning. In: Seel, M. (ed.) Encyclopedia of the Sciences of Learning, pp. 2325–2328. Springer, Boston (2012). https://doi.org/10.1007/978-1-4419-1428-6
18. Larsen, J.T., McGraw, A.P., Cacioppo, J.T.: Can people feel happy and sad at the same time? J. Pers. Soc. Psychol. **81**(4), 684 (2001)
19. Meeker, W.Q., Escobar, L.A.: Statistical Methods for Reliability Data, 1st edn. Wiley, New York (1998)
20. Sebe, N., Cohen, I., Gevers, T., Huang, T.S.: Multimodal approaches for emotion recognition: a survey. In: Proceedings of SPIE, vol. 5670, pp. 56–67 (2005)
21. Lera, E., Garreta-Domingo, M.: Ten emotion heuristics: guidelines for assessing the user's affective dimension easily and cost-effectively. In: Proceedings of 21st BCS HCI Group Conference, vol. 2, pp. pp. 163–166 (2007)
22. Kleinbaum, D., Klein, M.: Survival Analysis: A Self-Learning Text, 3rd edn. Springer, New York (2012). https://doi.org/10.1007/978-1-4757-2555-1

23. Bernardo, J.M., Smith, A.F.M.: Bayesian Theory, 1st edn. Wiley, New York (1994)
24. Sahu, S.K., Dey, D.K., Aslanidou, H., Sinha, D.: A Weibull regression model with gamma frailties for multivariate survival data. Lifetime Data Anal. **3**(2), 123–137 (1997)
25. Gamerman, D., Lopes, H.F.: Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference, 2nd edn. Chapman & Hall/CRC, Boca Raton (2006)
26. Lunn, D.J., Thomas, A., Best, N., Spiegelhalter, D.: Winbugs - a Bayesian modelling framework: concepts, structure, and extensibility. Stat. Comput. **10**(4), 325–337 (2000)
27. Niederreiter, H.: Some current issues in quasi-Monte Carlo methods. J. Complex. **19**(3), 428–433 (2003)

Scoring Summaries Using Recurrent Neural Networks

Stefan Ruseti[1], Mihai Dascalu[1,2,3](✉), Amy M. Johnson[4], Danielle S. McNamara[4], Renu Balyan[4], Kathryn S. McCarthy[4], and Stefan Trausan-Matu[1,2,3]

[1] University Politehnica of Bucharest, Splaiul Independenței 313, 60042 Bucharest, Romania
{stefan.ruseti,mihai.dascalu,stefan.trausan}@cs.pub.ro
[2] Academy of Romanian Scientists, Splaiul Independenţei 54, 050094 Bucharest, Romania
[3] Cognos Business Consulting S.R.L., Bd. Regina Maria 32, Bucharest, Romania
[4] Institute for the Science of Teaching and Learning, Arizona State University, PO Box 872111, Tempe, AZ 85287, USA
{amjohn43,dsmcnama,renu.balyan,ksmccar1}@asu.edu

Abstract. Summarization enhances comprehension and is considered an effective strategy to promote and enhance learning and deep understanding of texts. However, summarization is seldom implemented by teachers in classrooms because the manual evaluation requires a lot of effort and time. Although the need for automated support is stringent, there are only a few shallow systems available, most of which rely on basic word/n-gram overlaps. In this paper, we introduce a hybrid model that uses state-of-the-art recurrent neural networks and textual complexity indices to score summaries. Our best model achieves over 55% accuracy for a 3-way classification that measures the degree to which the main ideas from the original text are covered by the summary . Our experiments show that the writing style, represented by the textual complexity indices, together with the semantic content grasped within the summary are the best predictors, when combined. To the best of our knowledge, this is the first work of its kind that uses RNNs for scoring and evaluating summaries.

Keywords: Automated summary evaluation · Recurrent neural network
Semantic models · Word embeddings

1 Introduction

Summarization is an effective strategy to promote and enhance learning and deep understanding of the subject matter among students [1, 2]. Summarizing a text allows readers to differentiate between relevant and irrelevant information within texts, integrate content with pre-existing knowledge, allowing for both better retention of the text content [3], as well as deeper comprehension of the material [4]. Earlier studies have indicated that summary writing helps students retain new information [1]. Summary strategies are also effective for different types of learners including native speakers [5], language learners [6], students with learning disabilities [7] and students with low literacy skills [8]. A meta-analysis indicated that summarization

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 191–201, 2018.
https://doi.org/10.1007/978-3-319-91464-0_19

enhanced comprehension in 18 out of 19 studies [9]. Further, summarization is particularly useful for lower-skilled readers [10].

Given the effectiveness of summarizing texts, our aim is to develop computer-based summarization strategy training and practice that parallels an existing implementation of self-explanation and comprehension strategy practice within the Interactive Strategy Training for Active Reading and Thinking (iSTART) [11]. iSTART was developed to train comprehension strategies that help students understand complex, informational texts. Previous research demonstrated the effectiveness of iSTART for middle school [11], high school [12, 13], and college students [14, 15]. Currently, iSTART includes lesson videos covering four summarization strategies (deletion, main ideas, replacement, and topic sentences; see [16]). The development of practice modules in which students practice writing and revising summaries in turn necessitates a Natural Language Processing (NLP) algorithm capable of scoring the quality of summaries. ITSs that leverage NLP can provide students immediate, individualized feedback on their constructed (i.e., written) responses. This feedback is indispensable to learners attempting to improve their literacy skills [17].

Although summarization practice has proven effectiveness, teachers can find it challenging to implement practice activities because evaluating student summaries requires a great deal of effort and time [18]. Automated methods for summary evaluation traditionally involve evaluating quality metrics such as readability, content, conciseness, coherence and grammar [19]. In recent years, the research community has been successful in developing various measures for evaluating summaries. Some of the automated summary evaluation tools include Recall-Oriented Understudy for Gisting Evaluation (ROUGE [20]), ParaEval, Summary Input similarity Metrics (SIMetrix [21], QARLA [22], and SEMantic similarity toolkit (SEMILAR [23]).

The purpose of this study is to investigate the use of one of the most recent machine-learning techniques – recurrent neural networks (RNNs) [24] for automated scoring of summaries. To the best of our knowledge, this is the first work of its kind that uses RNNs for scoring and evaluating summaries.

The next section describes existing solutions and approaches used in literature for automated summary evaluation, and general deep-learning methods. In Sect. 3, the corpus, scoring rubric, followed by the proposed solution along with a detailed architecture is discussed. Finally, we report the results and conclude with discussions and future scope of the work.

2 Related Work

Evaluation of summaries is generally classified as intrinsic or extrinsic [25]. Intrinsic evaluation measures the *text quality* of summaries assessed by human annotators for fluency, informativeness and coverage, or evaluates the *content* of the summary using cue-words, term-frequency and inverted document frequency, cohesion methods, and Latent Semantic Analysis (LSA) [26]. By contrast, extrinsic evaluation is mostly task based involving document categorization, question answering and information retrieval [27]. The work described here focuses on intrinsic summary evaluation. Some of the earliest works in intrinsic summary evaluation include evaluation of chemistry documents [28] and electronic news publications [29]. Both of the latter studies used small data sets of 200 to 250 documents for evaluation. However, some early research efforts in large-scale evaluation of text summarization include TIPSTER SUMMAC [30] and the Document Understanding Conference (DUC). Researchers contributing to DUC have claimed that at large scales, even simple manual summary evaluations of content coverage and linguistic traits (e.g., capitalization errors, incorrect word order, unrelated fragments joined into one sentence, unnecessarily repeated information, misplaced sentences) requires a few thousand hours of human efforts [31]. In addition, some studies [32–35] show that human evaluations can be unstable and inconsistent with low inter-annotator agreement.

2.1 Automated Summary Evaluation

Some initial efforts towards developing automated summary evaluation metrics used n-gram overlap [33, 36]. These studies were motivated by the machine translation evaluation metric BiLingual Evaluation Understudy (BLEU) [37]. ROUGE [20] is one of the first and most widely used recall-oriented metrics for summary evaluations. ROUGE compares inputted summaries with one or multiple human written gold-standard summaries. One of the disadvantages of ROUGE is that all n-grams are considered equally important when computing the final score. Hovy et al. [38] proposed another simple metric based on basic elements' overlap, which are represented by one or two words, depending on their syntactic role.

Saggion et al. [39] proposed three content-based similarity measures: *cosine similarity, unit overlap* (unigrams or bigrams), and *longest common subsequence* (LCS). However, they did not discuss how these measures correlated with human evaluation. Another novel semi-automated approach is the *pyramid method* [40] which identifies and compares expert summaries' content units (SCUs) with to-be-evaluated summaries.

Some researchers have used random indexing [41, 42], that reduces terms by considering synonyms, hence allowing greater variations in summaries. Others have used distribution-similarity measures such as Kullback–Leibler (KL) divergence and Jensen Shannon (JS) divergence [21, 43], textual entailment [44] and crowdsourcing based LSA [18] for evaluating summaries. However, relatively few studies have used machine-learning techniques for summary evaluation beyond the aforementioned regression-based approaches [45–47].

2.2 Deep Neural Networks and Summary Evaluation

A common architecture used for text representation consists of recurrent neural networks, in particular Long Short-Term Memory networks (LSTM) [48] and Gated Recurrent Unit (GRU) [49]. These networks are capable of "memorizing" information, thus being able to better represent longer segments of text, without the danger of vanishing/exploding gradients encountered in traditional, normal recurrent neural networks [50]. These types of networks have been successfully used in most NLP tasks [51].

Recurrent neural networks have been improved further by considering different networks for the forward and backward directions [52]. This is especially useful when dealing with long text segments, because not all words in the text will have the same weight (e.g., depending on the language, the ones at the end are in most cases more important than the ones at the beginning). When using two different networks, the output for each word is usually represented by the concatenation of the outputs from the two directions. This way, all the words in the text influence the output for a single word.

We could not find any work that uses deep-learning techniques such as RNNs in particular for evaluating and scoring summaries. As a result, in order to explore the performance and success of these latest techniques for summary scoring and evaluation, we performed several experiments using RNNs.

3 Method

3.1 Corpus Description

We collected a corpus of 636 summaries for 30 texts (range: 20–24 summaries per text) using the Amazon Mechanical Turk online research service. The 30 texts used for the summary corpus collection were attained from the California Distance Learning Project (CDLP)[1], with permission from the Sacramento County Office of Education. The CDLP texts are real, simplified news stories that can be used by low-literate adults to improve their comprehension skills. The texts cover life-relevant topics, such as health and safety, housing, family, and money. Each text was between four and eight paragraphs and ranged from 128 to 452 words ($SD = 73.9$ words). Flesch-Kincaid grade level was between 4th and 8th grade ($SD = 1.1$) for all texts. The participants read and summarized three texts, randomly selected from the full set of 30 texts. Most of the participants (210/214) completed the entire summary task, producing three summaries total, for three separate texts. However, summaries submitted by the four participants who did not complete the entire task were also included in the corpus.

3.2 Scoring Rubric

Two trained researchers scored the summaries in the corpus on two major dimensions: (a) main ideas and (b) accuracy of main ideas. Before applying the coding scheme, the researchers individually examined the original texts, identifying the main ideas from each. Through discussions, they finalized a list of main ideas for each text. During coding of the summaries, the trained coders referenced this list of main ideas. For the main ideas dimension, each summary was scored from 0 (none of the main ideas from the text are included in the summary) to 3 (all of the main ideas from the text are included in the summary). For the accuracy of main ideas dimension, each summary was scored from 0 (main ideas present in the summary are completely inaccurate, or no main ideas are present in the summary) to 3 (all the main ideas in the summary are accurate representation of the content from the text).

[1] www.cdlponline.org.

Two trained raters scored the three dimensions for all 636 summaries. Inter-rater agreement for the Main Idea dimension was $\text{kappa}_{\text{linear weighted}} = .67$, $r = .78$, 71% exact agreement, and 99% adjacent agreement. Agreement for the Accuracy of Main Ideas dimension was $\text{kappa}_{\text{linear weighted}} = .44$, $r = .52$, 76% exact agreement, and 91% adjacent agreement. Differences between the ratings from the two researchers were resolved through discussions.

The distribution of the scores for main ideas and accuracy of main ideas is presented in Table 1. Due to the highly unbalanced distribution of the accuracy of main ideas dimension, it was not included in our follow-up experiments. Moreover, all 14 examples with a score of 0 for the main ideas dimension were ignored as there were not sufficient test cases in order to train a classifier.

Table 1. Distribution of output classes.

Score	No. of summaries	
	Main ideas	Accuracy of main ideas
0	14	28
1	165	22
2	255	61
3	202	525

3.3 Network Architecture

The network receives as input the summary and the original text, represented with pretrained Glove [53] word embeddings of size 100, ignoring words that were not part of the vocabulary. A BiGRU Siamese architecture (Fig. 1) was used to share network weights for the summary and the whole text. Max-pooling is performed on the forward-backward concatenated outputs from each cell. This results in two *2 * d* vectors (where *d* is the size of the GRU cell), representing the summary and the text. These two vectors are concatenated ("concat" operator from Fig. 1) and passed through two fully-connected layers (FCN module from Fig. 1).

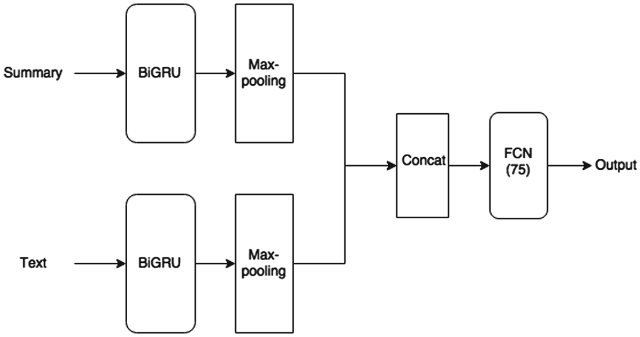


Fig. 1. Siamese recurrent network architecture.

The network produces a real number between 0 and 1, whereas our dataset has 3 output classes. Hence, we represented this task as a linear regression. To avoid force-fitting the network to the boundaries of this interval for the two extreme classes, the output score was multiplied by 4, resulting in a (0, 4) interval. When using the sigmoid activation function as output, it a good practice to avoid values close to 0 and 1 because the gradient for these flat regions is close to 0, making the training process difficult. Therefore, we split this domain and reassigned the predicted classes (i.e., 1, 2, and 3) as shown in Fig. 2, where all the three classes have almost equal range in the global interval.

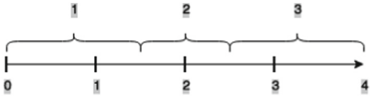


Fig. 2. Regression output.

As a baseline, we tested various complexity indices computed with the *ReaderBench* framework [54], which provides indices related to express the writing style of the text, instead of its content. From the available index categories, we extracted surface, syntax, word complexity, co-reference, connectives, cohesion, semantic dependencies, and word lists indices. Indices with low linguistic coverage (more than 20% of the values were missing) were removed and remaining indices were checked for multi-collinearity (Pearson $r \geq .9$). This cleaning process resulted in 191 features. These features were used to train two different models: a) individually within a 2-layered fully-connected network, and b) together within the recurrent network, as shown in Fig. 3. In both the cases, we tested two ways of using the complexity indices as input to the network: the difference between the two feature vectors (summaries and text) and the concatenation. Difference (marked as "diff" in Fig. 3) refers to the mathematic operator and is useful to highlight discrepancies between each feature or embedding dimension.

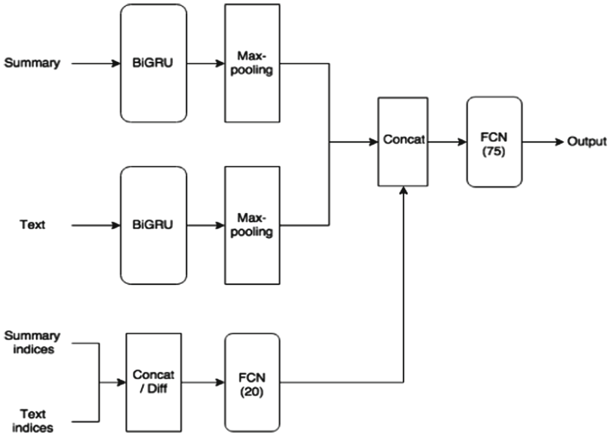


Fig. 3. Hybrid architecture of BiGRU, combined with *ReaderBench* textual complexity indices.

4 Results

As there were multiple summaries available for each text, the data were split into training and test sets (80–20). There were no common texts in the two partitions in order to avoid overfitting. The reported accuracies in Table 2 for each corresponding model were computed by averaging accuracy over three runs. The results in Table 2 indicate that the concatenation of feature vectors, despite needing more weights for the training process, works better and achieves better accuracy than the difference operator. This shows that important information is lost when the difference between the feature vectors is computed.

Table 2. Cell size and accuracy of models.

Model	Cell size	Accuracy (%)
Indices (difference)	-	41.90
Indices (concatenate)	-	37.14
Siamese	50	50.47
Siamese	100	50.15
Siamese + indices (difference)	50	47.30
Siamese + indices (difference)	100	53.34
Siamese + indices (concatenate)	**100**	**55.24**

In addition, we can observe from results in Table 2 that the complexity indices by themselves have the lowest accuracy, followed by the Siamese BiGRU network when used separately. The highest accuracy was obtained when combining the Siamese BiGRU network with the textual complexity indices from *ReaderBench*. This shows that both semantic features and writing style are important for summary evaluation.

5 Conclusions

This paper introduces a state-of-the-art model based on recurrent neural networks and textual complexity indices to evaluate and score summaries. To the best of our knowledge, this is the first work of its kind and the obtained accuracies of more than 55% is encouraging, given the size of the dataset. Moreover, our experiments show that the semantic content of the summary is more important than the writing style represented by the *Readerbench* textual complexity indices. However, replications with larger corpora should be conducted to support this conclusion.

Follow-up studies will also include an *attention* mechanism proven to be successful when comparing two or more text fragments by weighting the words with values computed based on the remainder of the text [55]. This mechanism is primarily used in question answering, but it can also be applied to summarization tasks by comparing the summary with the original text. However, the added weights may render the network too complex for this dataset, therefore reducing accuracy. In addition, the results might be improved by adjusting the hyper-parameters of the network using a grid-search method that performs cross-validations on the training set.

In sum, there are multiple ways in which this work can be validated and improved upon. However, this study demonstrates important promise in the use of recurrent neural networks to assess the quality of natural language.

Acknowledgment. This research was partially supported by the README project "Interactive and Innovative application for evaluating the readability of texts in Romanian Language and for improving users' writing styles", contract no. 114/15.09.2017, MySMIS 2014 code 119286, the 644187 EC H2020 RAGE project, the FP7 2008-212578 LTfLL project, the Department of Education, Institute of Education Sciences - Grant R305A130124, as well as the Department of Defense, Office of Naval Research - Grants N00014140343 and N000141712300.

References

1. Spirgel, A.S., Delaney, P.F.: Does writing summaries improve memory for text? Educ. Psychol. Rev. **28**, 171–196 (2016)
2. van Dijk, T.A., Kintsch, W.: Strategies of Discourse Comprehension. Academic Press, New York (1983)
3. Rinehart, S.D., Stahl, S.A., Erickson, L.G.: Some effects of summarization training on reading and studying. Read. Res. Q. **21**, 422–438 (1986)
4. Wade-Stein, D., Kintsch, E.: Summary Street: Interactive Computer Support for Writing (2004). http://www.tandfonline.com/doi/abs/10.1207/s1532690xci2203_3
5. Leopold, C., Sumfleth, E., Leutner, D.: Learning with summaries: effects of representation mode and type of learning activity on comprehension and transfer. Learn. Instr. **27**, 40–49 (2013)
6. Chiu, C.-H.: Enhancing reading comprehension and summarization abilities of EFL learners through online summarization practice. J. Lang. Teach. Learn. **5**(1), 79–95 (2015)
7. Rogevich, M.E., Perin, D.: Effects on science summarization of a reading comprehension intervention for adolescents with behavior and attention disorders. Except. Child. **74**, 135–154 (2008)
8. Perin, D., Lauterbach, M., Raufman, J., Kalamkarian, H.S.: Text-based writing of low-skilled postsecondary students: relation to comprehension, self-efficacy and teacher judgments. Read. Writ. **30**, 887–915 (2017)
9. Graham, S., Hebert, M.: Writing to read: a meta-analysis of the impact of writing and writing instruction on reading. Harv. Educ. Rev. **81**, 710–744 (2011)
10. Gil, L., Bråten, I., Vidal-Abarca, E., Strømsø, H.I.: Summary versus argument tasks when working with multiple documents: Which is better for whom? Contemp. Educ. Psychol. **35**, 157–173 (2010)
11. McNamara, D.S., O'Reilly, T., Rowe, M., Boonthum, C., Levinstein, I.: iSTART: a web-based tutor that teaches self-explanation and metacognitive reading strategies. In: Reading Comprehension Strategies: Theories, Interventions, and Technologies, pp. 397–420 (2007)
12. Jackson, G.T., McNamara, D.S.: Motivation and performance in a game-based intelligent tutoring system. J. Educ. Psychol. **105**, 1036–1049 (2013)
13. Snow, E.L., Jackson, G.T., McNamara, D.S.: Emergent behaviors in computer-based learning environments: computational signals of catching up. Comput. Hum. Behav. **41**, 62–70 (2014)
14. 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. J. Educ. Comput. Res. **32**, 185–208 (2005)

15. O'Reilly, T., Sinclair, G.P., McNamara, D.S.: iSTART: A web-based reading strategy intervention that improves students' science comprehension. In: IADIS International Conference Cognition and Exploratory Learning in Digital Age, pp. 173–180 (2004)
16. Johnson, A.M., Guerrero, T.A., Tighe, E.L., McNamara, D.S.: iSTART-ALL: confronting adult low literacy with intelligent tutoring for reading comprehension. In: André, E., Baker, R., Hu, X., Rodrigo, M.M.T., du Boulay, B. (eds.) AIED 2017. LNCS (LNAI), vol. 10331, pp. 125–136. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-61425-0_11
17. McNamara, D.S., Crossley, S.A., Roscoe, R.: Natural language processing in an intelligent writing strategy tutoring system. Behav. Res. Methods **45**, 499–515 (2013)
18. Li, H., Cai, Z., Graesser, A.C.: Computerized Summary Scoring: Crowdsourcing-Based Latent Semantic Analysis (2017). http://link.springer.com/10.3758/s13428-017-0982-7
19. Mani, I.: Automatic Summarization. John Benjamins Publishing, Amsterdam (2001)
20. Lin, C.Y.: Rouge: a package for automatic evaluation of summaries. In: Proceedings of Workshop Text Summarization Branches Out (WAS 2004), pp. 25–26 (2004)
21. Louis, A., Nenkova, A.: Automatically assessing machine summary content without a gold standard. Comput. Linguist. **39**, 267–300 (2013)
22. Amigó, E., Gonzalo, J., Penas, A., Verdejo, F.: QARLA: a framework for the evaluation of text summarization systems. In: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, pp. 280–289 (2005)
23. Rus, V., Lintean, M., Banjade, R., Niraula, N., Stefanescu, D.: SEMILAR: the semantic similarity toolkit. Assoc. Comput. Linguist. **2013**, 163–168 (2013)
24. Elman, J.L.: Finding structure in time. Cogn. Sci. **14**, 179–211 (1990)
25. Spärck Jones, K., Galliers, J.R.: Evaluating Natural Language Processing Systems, An Analysis and Review. Springer Science & Business Media, Heidelberg (1996). https://doi.org/10.1007/BFb0027470
26. Steinberger, J., Jezek, K.: Evaluation measures for text summarization. Comput. Inf. **28**, 1001–1025 (2009)
27. Jing, H., Barzilay, R., McKeown, K.R., Elhadad, M.: Summarization evaluation methods: experiments and analysis. In: AAAI Symposium on Intelligent Summarization, pp. 51–59 (1998)
28. Edmundson, H.P.: New methods in automatic extracting. J. ACM **16**, 264–285 (1969)
29. Brandow, R., Mitze, K., Rau, L.F.: Automatic condensation of electronic publications by sentence selection. Inf. Process. Manag. **31**, 675–685 (1995)
30. Mani, I., House, D., Klein, G., Hirschman, L., Firmin, T., Sundheim, B.: The TIPSTER SUMMAC text summarization evaluation. In: 9th Conference on EACL, p. 77. Association for Computational Linguistics, Morristown (1999)
31. Over, P., Yen, J.: An Introduction to DUC-2003 Intrinsic Evaluation of Generic News Text Summarization Systems (2003). http://www-nlpir.nist.gov/projects/duc/pubs/2003slides/duc2003intro.pdf
32. Donaway, R.L., Drummey, K.W., Mather, L.A.: A comparison of rankings produced by summarization evaluation measures. In: NAACL-ANLP 2000 Workshop on Automatic summarization, pp. 69–78. Association for Computational Linguistics (2000)
33. Lin, C.-Y., Hovy, E.: Manual and automatic evaluation of summaries. In: Proceedings of ACL02 Workshop on Automatic Summarization, vol. 4, pp. 45–51 (2002)
34. Rath, G.J., Resnick, A., Savage, T.: The formation of abstracts by the selection of sentences. J. Am. Soc. Inf. Sci. Technol. **12**, 139–141 (1961)
35. van Halteren, H., Teufel, S.: Examining the consensus between human summaries. In: Proceedings of the HLT-NAACL 2003 on Text Summarization Workshop, pp. 57–64. Association for Computational Linguistics, Morristown (2003)

36. Lin, C.-Y., Hovy, E.: Automatic evaluation of summaries using N-gram co-occurrence statistics. In: Proceedings of 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology, NAACL 2003, pp. 71–78 (2003)
37. Papineni, K., Roukos, S., Ward, T., Zhu, W.-J.: BLEU: a method for automatic evaluation of machine translation. In: ACL 2002, p. 311. Association for Computational Linguistics, Morristown (2001)
38. Hovy, E., Lin, C.-Y., Zhou, L., Fukumoto, J.: Automated summarization evaluation with basic elements. In: Proceedings of 5th International Conference on Language Resources and Evaluation, pp. 899–902 (2006)
39. Saggion, H., Radev, D., Teufel, S., Lam, W.: Meta-evaluation of summaries in a cross-lingual environment using content-based metrics. In: Proceedings of International Conference on Computational Linguistics, pp. 849–855 (2002)
40. Nenkova, A., Passonneau, R.: Evaluating content selection in summarization: the pyramid method. In: Proceedings of HLT-NAACL 2004, pp. 145–152 (2004)
41. Kanerva, P., Kristofersson, J., Holst, A.: Random indexing of text samples for latent semantic analysis. In: Proceedings of 22nd Annual Conference of the Cognitive Science Society, vol. 1036, pp. 16429–16429 (2000)
42. Sahlgren, M.: Vector-based semantic analysis: representing word meaning based on random labels. In: ESSLI Workshop on Semantic Knowledge Acquistion and Categorization (2002)
43. Lin, C.-Y., Cao, G., Gao, J., Nie, J.-Y.: An information-theoretic approach to automatic evaluation of summaries. In: Proceedings of the Main Conference on Human Language Technology Conference of the North American Chapter of ACL, pp. 463–470. Association for Computational Linguistics, Morristown (2006)
44. Bhaskar, P., Pakray, P.: Automatic evaluation of summary using textual entailment. In: RANLP 2013, pp. 30–37 (2013)
45. De, A., Kopparapu, S.K.: An unsupervised approach to automated selection of good essays. In: 2011 IEEE Recent Advances in Intelligent Computational Systems, RAICS 2011, pp. 662–666. IEEE (2011)
46. Ellouze, S., Jaoua, M., Belguith, L.H.: Machine learning approach to evaluate multilingual summaries. In: Proceedings of the MultiLing 2017 Workshop on Summarization and Summary Evaluation Across Source Types and Genres, pp. 47–54 (2017)
47. Perez-breva, L., Yoshimi, O.: Model Selection in Summary Evaluation, pp. 0–12 (2002)
48. Hochreiter, S., Urgen Schmidhuber, J.: Long short-term memory. Neural Comput. **9**, 1735–1780 (1997)
49. Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. In: EMNLP 2014, pp. 1724–1734 (2014)
50. Bengio, Y., Simard, P., Frasconi, P.: Learning long-term dependencies with gradient descent is difficult. IEEE Trans. Neural Netw. **5**, 157–166 (1994)
51. Conneau, A., Kiela, D., Schwenk, H., Barrault, L., Bordes, A.: Supervised learning of universal sentence representations from natural language inference data. In: EMNLP 2017 (2017)
52. Graves, A., Schmidhuber, J.: Framewise phoneme classification with bidirectional LSTM and other neural network architectures. Neural Netw. **18**, 602–610 (2005)
53. Pennington, J., Socher, R., Manning, C.D.: GloVe: global vectors for word representation. In: Proceedings of 2014 Conference on Empirical Methods in Natural Language Processing, pp. 1532–1543 (2014)

54. Dascalu, M., Gutu, G., Ruseti, S., Paraschiv, I.C., Dessus, P., McNamara, D.S., Crossley, S.A., Trausan-Matu, S.: ReaderBench: a multi-lingual framework for analyzing text complexity. In: EC-TEL 2017, pp. 495–499 (2017)
55. Santos, C. dos, Tan, M., Xiang, B., Zhou, B.: Attentive Pooling Networks. CoRR, abs/1602.03609, no. 2, p. 4 (2016)

Changes in Emotion and Their Relationship with Learning Gains in the Context of MetaTutor

Jeanne Sinclair[1](✉), Eunice Eunhee Jang[1], Roger Azevedo[2], Clarissa Lau[1], Michelle Taub[2], and Nicholas V. Mudrick[2]

[1] Ontario Institute for Studies in Education, University of Toronto, Toronto, ON, Canada
{jeanne.sinclair,clarissa.lau}@mail.utoronto.ca,
eun.jang@utoronto.ca

[2] Department of Psychology, North Carolina State University, Raleigh, NC 27695, USA
{razeved,mtaub,nvmudric}@ncsu.edu

Abstract. Positive academic emotions are generally associated with positive learning experiences, while the opposite is true for negative emotions. This study examined changes in learners' emotional profiles as they participated in MetaTutor, a computer-based learning environment designed to foster self-regulated learning via study of the human circulatory system. Latent transition analysis was employed to determine distinct, parsimonious emotional profiles over time. Learners are shown to move systematically among three profiles (positive, bored/frustrated, and moderate) in fairly predictable patterns. Of these, boredom is the most pressing concern given the relatively small chance of moving from boredom to a different emotional profile. Students' learning gains were also significant predictors of emotional transitions. The findings suggest the need for timely intervention for learners who are on the verge of negative emotional trajectories, and the complex relationship between learning gains and emotions. In addition, latent transition analysis is demonstrated as a potentially useful technique for analyzing and utilizing multivariate panel data.

Keywords: Computer-based learning environments · Academic emotions
Latent transition analysis

1 Study Objective, Background, and Framework

Students' emotions can impact their learning in computer-based learning environments (CBLEs), which in turn can have a reciprocal effect on emotional regulation (Azevedo et al. 2013; Calvo and D'Mello 2011). However, research on learner emotions is still in its infancy partly due to the conceptual and methodological challenges in operationalizing emotional constructs and measuring their dynamic changes during interactions with CBLEs (Azevedo et al. 2016). The goal of the present study was to determine the latent characteristics of a parsimonious model of learner emotions and further examine changes in learners' emotional profiles as they engage with an intelligent tutoring

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 202–211, 2018.
https://doi.org/10.1007/978-3-319-91464-0_20

system. In addition, we investigated the relationship between changes in emotional states and learning gains.

Learning-related emotions, also known as academic or achievement emotions (Pekrun 2006), are unique expressions of affect in the context of learning-related activities. These include process-related emotions such as enjoyment and boredom, and outcome-related emotions related to success such as hope and pride, or related to failure such as anxiety, hopelessness, and shame. Engagement, curiosity, and pride have been shown to be positively associated with learning, while anxiety, anger, boredom, and hopelessness have the opposite tendency (Pekrun et al. 2011). Beyond cognition, academic emotions are also associated with goal orientation. For example, students who are motivated to develop competence in a domain (mastery-oriented) are more likely to experience enjoyment, and less likely to feel bored or angry. Students who are motivated to demonstrate competence (performance-prove) or who feel a need to avoid being deemed incompetent (performance-avoid) are more likely to feel anxious, hopeless, and ashamed (Jang et al. 2017).

Students experience a range of emotions as they engage in complex learning in CBLEs, including engagement/flow, confusion, boredom, and frustration (D'Mello and Graesser 2012). Emotional states can fluctuate drastically, yet over time transitions are systematic when students are cognitively challenged (D'Mello and Graesser 2011). Positive emotions in CBLEs, however, may not always strongly correlate with learning gains (Jarrell et al. 2016). Confusion or cognitive dissonance, once resolved, can be associated with deep learning; however, inability to overcome frustration may result in less effective learning (Craig et al. 2004). Baker et al. (2010) caution specifically about boredom, as boredom appears to be quite persistent and prolonged periods of boredom are associated with poorer learning outcomes.

Affect is a key parameter that can improve the predictive accuracy and efficiency of intelligent tutoring systems. The bulk of scholarship pertaining to modeling emotions in CBLEs has applied statistical techniques such as Hidden Markov Modeling (e.g., Afzal and Robinson 2006), Gaussian Process Classification (e.g., Kapoor et al. 2007), Bayesian network models (e.g., Sam et al. 2012), or Dynamic Decision Network (e.g., Conati and Maclaren 2009). Latent transition analysis (LTA) is an emerging learner modeling approach to longitudinal data. In the present study, learners' emotional trajectories in the context of MetaTutor were modeled through the application of LTA. Our focus on the influence of learning gains on these emotional trajectories, rather than vice versa, is guided by previous research regarding the persistent impact of negative academic emotions on motivation and readiness to learn (e.g., Carver and Harmon-Jones 2009).

2 Methodology

2.1 MetaTutor Participants

This study is set in the context of MetaTutor, an intelligent tutoring system that operationalizes theoretical models of self-regulated learning (SRL) while advancing students' understanding of scientific concepts. MetaTutor offers a variety of strategies for students to plan, monitor, and reflect upon their learning, which are core elements of SRL

(Azevedo et al. 2013; Lau et al. 2017). Students are prompted to set learning goals and engage in cognitive and metacognitive SRL strategies. Based on students' quiz scores and metacognitive judgments, MetaTutor provides feedback and options for next steps. One hundred ninety-four students recruited from three North American universities participated in this study (53% male, mean age 20.46 (*SD* = 2.92)).

2.2 Instruments and Data

Emotions and Values Questionnaire. Metatutor participants were prompted every 14 min to complete the Emotions and Values [EV] questionnaire based on Pekrun's (2006) activity emotions. This frequency is designed to glean learning processes in addition to learning outcomes. The EV is a self-report measure with the prompt, "Right now I feel…" followed by individual emotions and a Likert scale (1 for Strongly disagree to 5 for Strongly agree). While the current study addresses learners' responses regarding five emotions (enjoyment, curiosity, pride, boredom, and frustration), the EV also addresses anxiety, shame, hopelessness, surprise, contempt, sadness, and feelings of eureka and neutrality. After initial analysis, these were excluded from the current study after finding that they lacked variation, given that our goal was to find the most parsimonious set of emotions that distinguish the latent classes.

Learners' sessions differed in overall length. For the present study, learners' latent emotion classes, based on five emotion self-reports, were modeled at three timepoints (TP): 2, 4, and 6. TP1 represented the initial EV measure, prior to learners' engagement with Metatutor, so it was not included. We limited the analysis to these three timepoints for computational efficiency and missingness after TP6. Participants who did not complete TP2 were excluded resulting in a final sample of n = 190.

Raw Learning Gains. Prior to participating in MetaTutor students completed a 25-item pretest on the human circulatory system (*M* = 17.27, *SD* = 4.45), and after two MetaTutor sessions they completed a 25-item posttest (*M* = 20.60, *SD* = 4.20). The order of pre- and post-tests was counter-balanced, and learning gains were statistically significant for the entire sample (t = 12.71, p < .001), indicating improvement over time. Raw learning gains were determined by subtracting pre- from post-test scores; negative values were retained. The range of learning gains was −6 to 12, *M* = 3.33, *SD* = 3.65.

2.3 Latent Transition Analysis

Latent transition analysis (LTA), a longitudinal outgrowth of latent class analysis (LCA), models transitions between latent profiles or states over time (Collins and Lanza 2013). While LCA identifies individuals who share similar latent characteristics and classifies them into exhaustive and mutually exclusive classes, LTA goes further to estimate transitional probabilities of these latent classes across time. The first step is to determine the composition of the latent classes. Then, learners are tracked over time to understand if they remain in the same profile or if their profiles change.

In this study, LTA modeling was performed using MPlus 7. Multiple latent models were tested to determine the appropriate number of latent classes (Collins and Lanza

2013). The best fitting model was chosen by increasing the number of classes serially and examining model-fit indicators based on lower values and the "elbow" formed by Bayesian Information Criterion (BIC) and Akaike's Information Criterion (AIC), the distinguishability of classes, homogeneity of item conditional probabilities within classes, entropy values, and interpretability. Raw learning gains were then included as a covariate to examine their influence on the emotional profiles' transitions over time.

This methodology is similar to Baker et al.'s (2010) study of persistence which examined the likelihood of transition compared to the base rate of each cognitive-affective state. The present study uses transitional probabilities between 0 and 1 to describe movement among states and covariates to understand the influence of learning gains on changes in emotion.

3 Results and Discussion

Three LCA models were applied to the data to determine the number of latent emotion classes at each time point. A three-class solution emerged as the best fit for the data, as indicated by lower AIC and BIC values in Fig. 1.

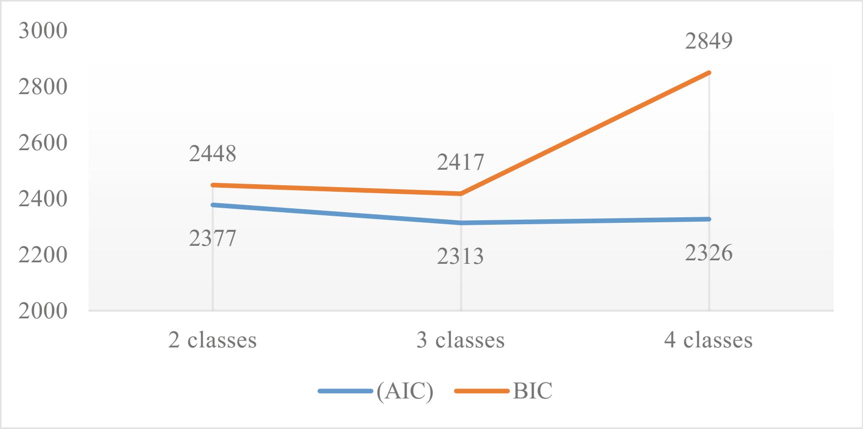


Fig. 1. Model fit indices (AIC = Akaike's Information Criterion; BIC = Bayesian Information Criterion)

The composition of the three classes is demonstrated in Fig. 2. The positive emotions class consists of learners who demonstrated enjoyment, pride, and curiosity, and less frustration and boredom. The second class exhibited boredom with some frustration, and low endorsement of positive emotions. Lastly, the third class indicates moderate endorsement of all emotions, with relatively lower frustration. The three classes are somewhat consistent with Jarrell et al.'s (2016) finding of three emotional subgroups in the context of a CBLE: one positive, one negative, and one low emotion. However, boredom and frustration did not separate, as might have been predicted by Baker et al. (2010).

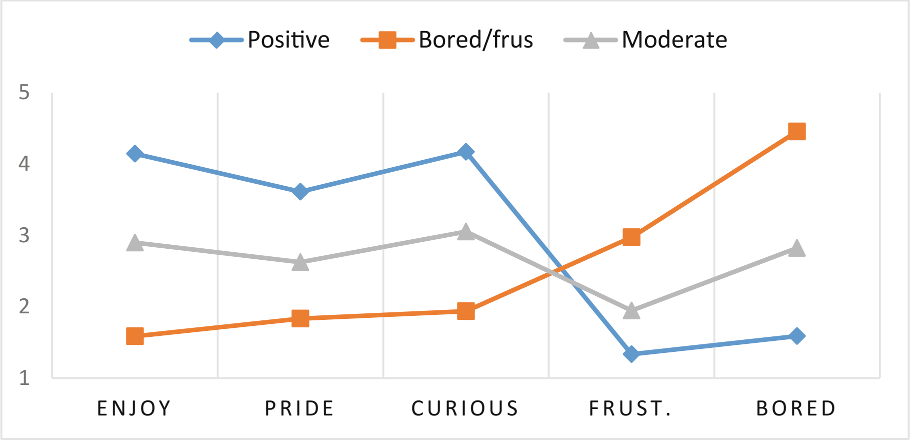


Fig. 2. Composition of the three latent classes.

Subsequently, latent transition modeling was applied to the longitudinal data. In LTA, the three classes were constrained to have the same composition (estimated means of each emotion) over the three timepoints. Table 1 indicates the probability of latent class pattern over the three timepoints. Patterns representing zero learners were omitted

Table 1. Prevalence of each latent class pattern based on posterior probabilities. Profiles with zero members (e.g., Positive → Bored/frust → Positive are excluded)

TP2	TP4	TP6	Estimated count	Estimated proportion
Positive	**Positive**	**Positive**	**28**	**.15**
Positive	Positive	Bored/frust	1	.01
Positive	Positive	Moderate	7	.04
Positive	Moderate	Positive	1	.01
Positive	Moderate	Bored/frust	2	.01
Positive	Moderate	Moderate	28	.15
Bored/frust	Positive	Positive	1	.01
Bored/frust	Bored/frust	Positive	1	.01
Bored/frust	**Bored/frust**	**Bored/frust**	**20**	**.11**
Bored/frust	Bored/frust	Moderate	2	.01
Moderate	Positive	Positive	4	.02
Moderate	Positive	Moderate	1	.01
Moderate	Bored/frust	Bored/frust	24	.13
Moderate	Bored/frust	Moderate	1	.01
Moderate	Moderate	Bored/frust	18	.09
Moderate	**Moderate**	**Moderate**	**51**	**.27**
Total			**190**	

Note: Bolded trajectories remain in the same profile across all three TPs.

from the table. Bolded rows indicate remaining in the same class over the three timepoints, which represents the most common pattern (over half of the sample).

Other common patterns include moving from Positive to Moderate, and from Moderate to Bored/frustrated, but no learners moved from the two extremes (Positive and Bored/frustrated) in a single transition. This is of interest because learners appear to move stepwise or gradually between emotional profiles, contrary to the idea that these transitions may occur quickly (cf. Graesser et al. 2014).

To better understand learners, it is important to understand the probabilities of transitioning from one class to another over time. Table 2 demonstrates the probability of remaining in the same class or transitioning to another class across the three timepoints. As seen in Table 1, learners were more likely than not to remain in the same class (bolded values in Table 2). For learners who did move to another emotional profile, positive emotions tend to transition to moderate emotions, while moderate emotions moved into boredom. Positive emotions were not at all likely to directly lead to Boredom/frustration. The class exhibiting boredom was most likely of all classes to remain in the same class (.95 in first transition and .86 in second transition). This is consistent with Baker et al.'s (2010) finding of the persistence of boredom.

Table 2. Transitional probabilities over time

	Positive emotions	Bored, some frustration	Moderate, less frustration
Moving from TP2…	*…to TP4*		
Positive	**.54**	<.01	.46
Bored, some frus.	.05	**.95**	<.01
Moderate	.05	.27	**.68**
Moving from TP4…	*…to TP6*		
Positive	**.72**	.04	.24
Bored, some frus.	.03	**.86**	.11
Moderate	.02	.21	**.77**

Students' learning gains (LGs) were entered into the LTA as a covariate through multinomial logistic regression using the Moderate class as the reference class. Table 3 indicates the log odds and significance of learning gains as a predictor influencing the transitional probabilities. LG was a significant predictor of transitions from TP2 Positive to TP4 Positive, and a significant predictor of moving from the Bored/Frustrated class at TP2 to Positive in TP4, which is notable because only one learner did actually move from Bored/frustrated to Positive. In addition, LG is a significant predictor of moving from Positive to Bored/frustrated. Between the second two timepoints, LG remained a significant predictor of remaining in the Positive class, as well as from Bored/frustrated to positive.

Table 3. Log odds and *p*-values of LG predicting emotional transitions

	Moving to…		
	Positive emotions	Bored, some frustration	Moderate, less frustration
Moving from TP2…	*…to TP4*		
Positive emotions	.88**	−.98**	Ref
Bored, some frustration	.93**	−.64	Ref
Moderate, less frustration	1.43	.57	Ref
Moving from TP4…	*…to TP6*		
Positive emotions	.50*	.34	Ref
Bored, some frustration	45.89**	45.96	Ref
Moderate, less frustration	.52	.33	Ref

**p < .01, * p < .05

Using the Latent Transition Analysis Calculator in MPlus, the relationship between LG and emotional states was further examined at three points on the distribution: the sample mean (*M* = 3.33) and ± *SD* 1.5. The results are found in Table 4, demonstrating the difference in probability for each latent class pattern contingent on LG. Table 4 is useful for interpretation because it does not require a reference class. Of importance is that Table 4 does not provide information about the probability of class membership overall; rather, it indicates the influence of different LG levels on whether a learner's emotional profile pattern, over time, transitions among the three classes. The first column of Table 4 indicates the latent class pattern, first with only time point 2, then with time points 2 and 4, and finally with all three timepoints. The finding that there is no influence of LG on class membership at TP2 (first three rows) is meaningful and supports the validity of this analysis, as learning had not yet begun.

In the first set of transitions (TP2–TP4), an increase in LG is associated (by looking across each row) with a small decrease in the probability of remaining in the Positive class and a greater likelihood of moving to Moderate. In general, increase in LG is associated with little effect on remaining in Boredom/Frustration from TP2–TP4, as the three values in the row are quite similar. However, movement from Moderate appears to be influenced by LG, with lower LG more likely to remain in Moderate, and higher LG slightly increasing the likelihood of moving to Boredom.

Looking at the last set of transitions in Table 4 (TP2–TP6), learners who remained Positive at TP2 and TP4 did not have large differences in which class they transitioned to at TP6, based on LGs; however, greater LG is associated with moving to Moderate rather than remaining Positive. For Positive at TP2 and Moderate for TP4, greater LGs were associated with a greater chance of remaining in Moderate. Interestingly, for students who were Bored TP2 and TP4, greater LGs were associated with staying in Bored at TP6; this was similar to the Moderate → Bored → Bored trajectory. Finally, for learners who were Moderate at TP2 and TP4, those with greater LGs were more likely to remain in Moderate at TP6, and less likely to move to Bored.

Table 4. Probability of latent class patterns contingent on learning gains (LGs), for transitions prevalent in the data per Table 1

Latent class pattern	LG = −2.15	LG = 3.33	LG = 8.81
TP 2			
Pos	.35	.35	.35
Bor	.13	.13	.13
Mod	.52	.52	.52
TPs 2–4			
Pos → Pos	.60	.54	.48
Pos → Mod	.40	.46	.52
Bor → Bor	.96	.95	.95
Mod → Bor	.21	.28	.29
Mod → Mod	.79	.71	.51
TPs 2–6			
Pos → Pos → Pos	.74	.72	.70
Pos → Pos → Mod	.20	.24	.27
Pos → Mod → Mod	.69	.78	.85
Bor → Bor → Bor	.00	.96	.99
Mod → Bor → Bor	.00	.96	.99
Mod → Mod → Bor	.29	.20	.13
Mod → Mod → Mod	.69	.78	.85

Note: Pos = positive emotions, Bor = bored, some frustration, Mod = moderate, less frustration.

The emerging pattern from the LG covariate analysis is that higher learning gains were associated with maintenance of Moderate emotions at all TPs, maintenance of Boredom/frustration at all three TPs, and negative trajectories (e.g., Positive (TP2) → Moderate (TP4)). Lower learning gains are associated with Positive maintenance, and one negative trajectory, Moderate → Moderate → Bored.

4 Implications, Limitations, Conclusion

Learners' transitions among different emotion profiles showed some consistent patterns. More than half of the sample remained in the same latent class across all three TPs, and learners who began in Bored/frustrated had the highest probability of remaining in that state, which is consistent with Baker et al.'s (2010) finding of the persistence of boredom. Learners who did transition from one emotional profile to another were not likely to "jump" from the two extremes (Positive and Bored/frustrated). Indeed, they were even unlikely to move between the two extremes during the course of their MetaTutor sessions. These findings suggest that the patterns of emotional change are somewhat predictable.

A positive implication is that a window of time (when learners are in the Moderate state) could allow pedagogical agents to intervene with learners before they descend into boredom, a state which appears to retain learners, similar to Baker et al.'s (2010) finding on the persistence of boredom. Effective interventions could include prompts to take a short break (perhaps accompanied by a video, puzzle, or prompt for physical exercise), or a prompt to reflect in writing on learning progress, or perhaps allowing the learner to customize the pace and delivery of learning content.

In terms of learning gains, the transition represented by the largest proportion of the sample, Moderate (TP2) → Moderate (TP4), were more likely to move to Moderate (TP6) if they had higher learning gains. Similarly, students who were Bored/frustrated throughout also had higher learning gains. These students may be regulating their emotions as they learn challenging content, vis-à-vis students who remained Positive, who were more likely to have lower learning gains. Learners who were Moderate until TP4 were more likely to move to Bored if they had lower learning gains. Thus planning a "window" for intervention may consider not just which emotional states learners are moving to/from, but also the timing of those transitions.

Unexpectedly, negative emotional trajectories were associated with positive learning gains. This could be related to extended engagement while learning complex material and associated feelings of fatigue. Also, given Baker et al.'s (2010) findings about the relative benefit of frustration over boredom, that our best fitting latent class model had a single combined frustration/boredom class may impact this result. In other words, because learners in this class exhibited both frustration and boredom, which also is predicted by D'Mello and Graesser (2012), frustration possibly helped them persevere.

Emotion is a complex construct that requires a multimodal approach to conceptualizing and measuring its dynamic nature (Azevedo et al. 2016). In the present study, students' emotions were measured through self-reports which are subject to various sources of response bias. Despite such potential limitations, the results from the present study offer some significant insights into the emotional trajectories students experience during learning in CBLEs.

References

Afzal, S., Robinson, P.: A study of affect in intelligent tutoring. In: Workshop on Modeling and Scaffolding Affective Experiences to Impact Learning, vol. 57, pp. 27–53 (2006)

Azevedo, R., Harley, J., Trevors, G., Feyzi-Behnagh, R., Duffy, M., Bouchet, F., Landis, R.S.: Using trace data to examine the complex roles of cognitive, metacognitive, and emotional self-regulatory processes during learning with multi-agent systems. In: Azevedo, R., Aleven, V. (eds.) International Handbook of Metacognition and Learning Technologies, pp. 427–449. Springer, Amsterdam (2013). https://doi.org/10.1007/978-1-4419-5546-3_28

Azevedo, R., Martin, S.A., Taub, M., Mudrick, N.V., Millar, G.C., Grafsgaard, J.F.: Are pedagogical agents' external regulation effective in fostering learning with intelligent tutoring systems? In: Micarelli, A., Stamper, J., Panourgia, K. (eds.) ITS 2016. LNCS, vol. 9684, pp. 197–207. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39583-8_19

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. Int. J. Hum.-Comput. Stud. **68**(4), 223–241 (2010)

Carver, C.S., Harmon-Jones, E.: Anger is an approach-related affect: evidence and implications. Psychol. Bull. **135**, 183–204 (2009)

Calvo, R.A., D'Mello, S. (eds.): New Perspectives on Affect and Learning Technologies. Springer, New York (2011). https://doi.org/10.1007/978-1-4419-9625-1

Collins, L.M., Lanza, S.T.: Latent Class and Latent Transition Analysis: With Applications in the Social, Behavioral, and Health Sciences. Wiley, Hoboken (2013)

Conati, C., Maclaren, H.: Empirically building and evaluating a probabilistic model of user affect. User Model. User-Adap. Inter. **19**(3), 267–303 (2009)

Craig, S.D., Graesser, A.C., Sullins, J., Gholson, B.: Affect and learning: an exploratory look into the role of affect in learning with AutoTutor. J. Educ. Media **29**(3), 241–250 (2004)

D'Mello, S., Graesser, A.: The half-life of cognitive-affective states during complex learning. Cogn. Emot. **25**(7), 1299–1308 (2011)

D'Mello, S., Graesser, A.: Dynamics of affective states during complex learning. Learn. Instr. **22**(2), 145–157 (2012)

Graesser, A.C., D'Mello, S.K., Strain, A.C.: Emotions in advanced learning technologies. In: International Handbook of Emotions in Education, pp. 473–493 (2014)

Jang, E.E., Lajoie, S.P., Wagner, M., Xu, Z., Poitras, E., Naismith, L.: Person-oriented approaches to profiling learners in technology-rich learning environments for ecological learner modeling. J. Educ. Comput. Res. **55**(4), 552–597 (2017)

Jarrell, A., Harley, J.M., Lajoie, S.P.: The link between achievement emotions, appraisals, and task performance: pedagogical considerations for emotions in CBLEs. J. Comput. Educ. **3**(3), 289–307 (2016)

Kapoor, A., Burleson, W., Picard, R.W.: Automatic prediction of frustration. Int. J. Hum. Comput. Stud. **65**(8), 724–736 (2007)

Lau, C., Sinclair, J., Taub, M., Azevedo, R., Jang, E.E.: Transitioning self-regulated learning profiles in hypermedia-learning environments. In: Proceedings of the Seventh International Learning Analytics & Knowledge Conference, pp. 198–202. ACM, March 2017

Pekrun, R.: The control-value theory of achievement emotions: assumptions, corollaries, and implications for educational research and practice. Educ. Psychol. Rev. **18**(4), 315–341 (2006)

Pekrun, R., Goetz, T., Frenzel-Anne, C., Petra, B., Perry, R.P.: Measuring emotions in students' learning and performance: the achievement emotions questionnaire (AEQ). Contemp. Educ. Psychol. **36**, 34–38 (2011)

Sam, Y.C., Ting, C.Y., Wong, C.O.: Extrapolating the role of affect into supporting conceptual change in scientific inquiry learning. In: 2012 International Conference on Information Retrieval & Knowledge Management (CAMP), pp. 158–161. IEEE (2012)

A Heuristic Approach for New-Item Cold Start Problem in Recommendation of Micro Open Education Resources

Geng Sun[1(✉)], Tingru Cui[1], Dongming Xu[2], Jun Shen[1],
and Shiping Chen[3]

[1] School of Computing and Information Technology, University of Wollongong,
Wollongong, Australia
{gsun, tingru, jshen}@uow.edu.au
[2] UQ Business School, The University of Queensland, Brisbane, Australia
D.Xu@business.uq.edu.au
[3] CSIRO Data61, Sydney, Australia
Shiping.Chen@data61.csiro.au

Abstract. The recommendation of micro Open Education Resources (OERs) suffers from the new-item cold start problem because little is known about the continuously published micro OERs. This paper provides a heuristic approach to inserting newly published micro OERs into established learning paths, to enhance the possibilities of new items to be discovered and appear in the recommendation lists. It considers the accumulation and attenuation of user interests and conform with the demand of fast response in online computation. Performance of this approach has been proved by empirical studies.

Keywords: Cold start · Open Education Resources · Adaptive micro learning
Heuristic recommendation · Learning path

1 Introduction

Along with various leading universities opening up access to their courses, open education resources (OERs) are becoming increasingly available. The rise of OERs gains large popularity in the entire higher and adult education sector, and these emerging learning paradigms have attracted many researchers' attention, from educational, social, and computational views [1]. However, because of the newness of the phenomenal education trend, its popularity has been limited due to the lack of personalized services so that current OER delivery often fails to meet comparatively diverse demands from both OER providers and learners. Moreover, these OERs are suggested to be consumed in a micro learning mode, which conforms to the characteristic of the modern e-society where mobile and pervasive computing becomes dominant.

A service-oriented system, Micro Learning as a Service (MLaaS), aims to deliver personalized OER with micro learning to satisfy learners' personal demands in real

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 212–222, 2018.
https://doi.org/10.1007/978-3-319-91464-0_21

time. It customizes adaptive micro learning contents as well as provides learning path identifications tailored for each individual learner. MLaaS consists of an offline computation and an online computation domain to provide recommendations jointly to improve computation performance and respond in the granularity of seconds. In this paper we will introduce a heuristic approach to overcome the new-item cold start problem in micro OER recommendation, which will be realized by the online computation of MLaaS.

2 Background

2.1 Micro Learning

Compared to traditional learning modes, now learners' overall efforts to go through an entire concept (or learning objective) will proceed in an intermittent way rather than a consecutive way. Hence, micro learning through OER (i.e. micro open learning) is becoming mainstream for next generation learners, who learn on the move, with easy access to the 'cloud' or Internet of Things [2]. Generally micro learning refers to short-term learning processes, which contains knowledge or learning content in small units. Typically, a micro learning activity is carried out through mobile devices within a time frame of 15 min [3]. As an emerging educational phenomenon, micro learning is more user-centric and it requires different learning schedules from on-campus learning or even standard e-learning and m-learning [4].

2.2 Micro Learning as a Service

Basically, the offline computation of MLaaS runs on a basis of compound transactions. When a learning activity launches, a compound transaction is generated associated with it, which can be therefore represented as: *{Learner Profile, Micro OER profile, Association},* where a learner profile and an OER profile are involved, linked by an association showing the learner's properties against the micro OER (Fig. 1).

The offline computation is responsible to transform implicit user behaviors into explicit information [5]. A knowledge base is in charge of the semantic construction and storage of the learner profile and OER representation [6]. The semantic construction of learner profiles enables MLaaS to consider individual learning styles, learner's context, application capabilities, and teaching materials structure, leading to a customization of the type and delivery format of learning information in response to the user. Similarly, an augmented micro OER ontology is also built [7].

The authors of [7] have proposed a comprehensive learner model which involves features that can impact and constrain the micro learning experiences and outcomes, and is enclosed in an ontological representation [8]. By taking advantage of the comprehensive learner model, the LearnerProfile can be broken down to: *{InternalFactors, ExternalFactors} = {IntellectualFactors & NonIntellectualFactors, ExternalFactors}*, where the internal factors can be classified into personal intellectual and non-intellectual factors, differentiated by whether a factor is related to a learner's

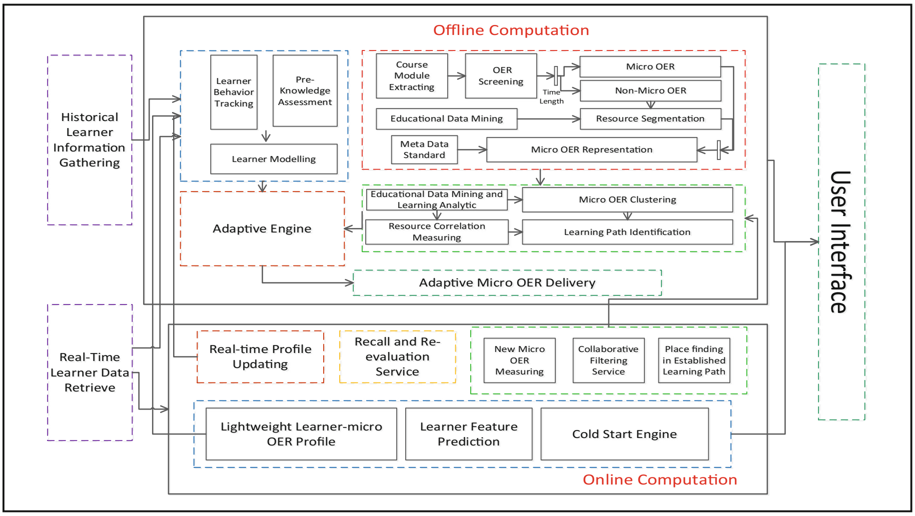

Offline Computation
Historical Learner Information Gathering
Learner Behavior Tracking
Pre-Knowledge Assessment
Learner Modelling
Course Module Extracting
OER Screening
Time Length
Micro OER
Non-Micro OER
Educational Data Mining
Resource Segmentation
Meta Data Standard
Micro OER Representation
Adaptive Engine
Educational Data Mining and Learning Analytic
Micro OER Clustering
Resource Correlation Measuring
Learning Path Identification
Adaptive Micro OER Delivery
User Interface
Real-Time Learner Data Retrieve
Real-time Profile Updating
Recall and Re-evaluation Service
New Micro OER Measuring
Collaborative Filtering Service
Place finding in Established Learning Path
Lightweight Learner-micro OER Profile
Learner Feature Prediction
Cold Start Engine
Online Computation

Fig. 1. Architecture of online and offline computation in MLaaS

cognitive and intelligence level or not. External factors come from the environmental and social-economical contexts.

Given the OER delivery has a ‘big data’ context, ideally there could be sufficient data sets to be used in data modeling and machine learning, and rule mining is compulsory to impute unknown values for online-prediction. The rules can be represented as:*{LearnerProfile, MicroOERProfile}* → *{Association}*.Technically rule mining and learner clustering operation runs throughout all the offline computation process.

As the core of MLaaS, the Adaptive Engine processes the results from all other services and transmits its output to the user interfaces straightforward.

However, the initial MLaaS system has insufficient information about the learners as well as new OERs without existing ratings, which leads to infeasible profile construction. This places the cold start problem as the central challenge of micro OER delivery. Consequently, MLaaS’s online computation is mainly in charge of the cold start problem, and also making up limitation in timeliness and renewal of the offline computation by retrieving real-time usage and keeping the comprehensive learner model and leaner-micro OER profiles up-to-date.

In [9] a solution to deal with the new-user cold start problem has been proposed. A top-N recommendation is adopted to provide learners a range of options to kick off their micro open learning journey. It takes a rule-based heuristic approach to generate candidate learning paths, and then the first micro OER in each learning paths is picked up to form the recommendation list. In this paper we will carry on the efforts put in the new-user cold start problem and follow the rules to come up with a new heuristic for the new-item cold start problem, namely inserting newly published micro OERS into well-established learning paths.

2.3 Research Problem

The cold start problem is generally triggered by three factors: new community, new item and new user. Both open learning and micro learning are comparatively novel to many people in the e-society. The followers of this novel trend, no matter new education pursuers or regular learners migrated from other online learning modes, newly join into this emerging community of e-society. Meanwhile, MLaaS faces the new-item cold start which is of great practical importance. This is because new OERs are kept publishing day after day, and effectively recommending them is essential for keeping the users continuously engaged [10]. For newly published micro OERs, it is central to their acceptance and popularity that it can be discovered and appear in recommendation list as soon as being released.

The learning demands and expectations of learners engaged in open learning are much more practical than conventional university students. They are mostly self-regulated so that it is totally flexible for them to decide when to join or quit the online course at their own willingness, and switch among courses frequently [11].

In addition, in micro open learning, the accumulation and attenuation of user interests and demands can be periodical and may vary in different patterns than other online activities [12]. It is very likely to see that a learner accesses OERs covering similar knowledge are-as again (and again) offered by different educational institutions. This cross-learning phenomenon can be attributed to purposes of reviewing or mutual supplementation, by comparing the ways of knowledge imparting as well as learning two or more micro OERs simultaneously [11, 12].

3 Semantic Representation of Micro OERs and Learners

3.1 Investigation of Micro OER Correlation

From the item-based perspective, the general ontology of OER is augmented to adapt the needs of micro learning, in which an annotation of a micro OER is self-describing with metadata exploring its educational parameters, such as typology, type of interaction, didactic model, and non-functional attributes. Each node in the augmented OER ontology indicates a micro OER chunk [7]. There is no completely independent chunk and each of them is part of a relational web rather than merely a conceptual object. This ontology explicitly classifies the OERs to recommend among a pedagogically defined set of distinctive main concepts, fed as the raw material in the reasoning process of MLaaS [13].

The investigation over ‘big’ open learning data comes up to the OER side. Among the massive OERs, three types of relations are mainly targeted:

- ConsistsOf is an inclusion relation. This relation can be generally found between two OERs or one OER and one micro OER. Two items with this relation are located in different hierarchies of the augmented micro OER ontology, the ancestor at the top and the descendant at the bottom. Let R denote an *OER* and MR denote a micro OER, and ConsistsOf (R_a, MR_b), or $MR_b \in R_a$ indicates that the original OER R_b(ancestor) is segmented into several micro OERs (descendants), and MR_b is one of them. Certainly, as a micro OER can be further subdivided, there can be ConsistsOf (MR_b, MR_c), or $MR_c \in MR_b$ provided that MR_c is a tinier micro OER derived from MR_b.

- RequiredSequence is a strong order between two items (OER or micro OER), where the former micro OER is necessary to be learnt before the latter one, due to course setting and educational consideration.
- RecommendedSequence is a weak order relation between two items (OER and micro OER), where the former micro OER is suggestive to be learnt before the latter one, according to the instructors' guidance, but it is not mandatory.
- Certainly, two items (OER or micro OER) can have no relation at all.

Both relations regarding sequence can be inherited by entities' descendants. For example, when there is a RecommendedSequence (R_1, R_2) indicating an OER R_1 is preferably learnt prior to R_2, then, for $MR_1 \in R_1$ *and* $MR_2 \in R_2$, there is a RecommendedSequence (MR_1, MR_2). The inheritance is also valid if the ConsistsOf relation is between two micro OERs.

3.2 Lightweight Learner-Micro OER Profile

Motivated by the cold start condition, the comprehensive learner model was simplified to a lightweight learner-micro OER profile [7]. It merely deals with necessary information for decision making in order to act on the initialization agilely.

The lightweight learner profile is managed by MLaaS with a static part and a dynamic part. The static part can be represented by a vector, which contains the demographic and educational information. By matching these two augmented ontologies, for item and user respectively, the dynamic part of a learner node is denoted as a pair, $L_j = \{MR_u, ML_j\}$, $L_j \in L$. Herein, the element MR_u denotes the u^{th} micro OER, which is a particular version of the micro OER ontology, as introduced in the previous subsection, and a three-dimensional element ML_j $\{P_{u,j}, TA_j, D_j\}$ is exclusive to j^{th} learner during the micro learning process. Herein, the element $P_{u,j}$ indicates the learner's preference, TA_j indicates the j^{th} learner's instant time availability, and D_j denotes the level of distraction in terms of the given learning environment and surroundings. Each of these three features proposed in the lightweight profile is associated with a confidence degree to reflect its subjective relevance. Whenever MLaaS gathers any information from the learner's learning process over OER, the learner profile will be updated regarding ML_j.

4 Insertion of New Micro OERs into Established Learning Path

4.1 Micro OER Screening and Rules

For each existing micro OER, once MLaaS has acquired its final preference value and confidence degree, those nodes, which do not meet the minimum requirement of confidence degree, is rejected by the system. When generating a list of recommended micro OERs, the ones with higher learners' interests are placed at the top. For two micro OERs MR_u and MR_w, their sequence is determined according to some heuristic rules which are defined in accordance with the extraction of three types of relations discussed in the Sect. 3.1. These rules are executed sequentially with priority. Herein, the first rule is deemed as a hard rule which should be strictly obeyed and the rest rules are soft rules which can be violated with educational consideration, from case to case.

1. If there is a RequiredSequence relation between these two micro OERs, the prerequisite one is placed above (refer to the Sect. 3).
2. If the preference regarding these two OERs, $P_{u,j}$, $P_{w,j}$, the former one is higher than the latter one, then the MR_u is above MR_w
3. If, in the absolute terms, the confidence degree $CD(P_{u,\ j})$ is high and the $CD(P_{w,j})$ is low, then the MR_u is above MR_w.
4. If there is a RecommendedSequence relation between these two micro OERs, the one which is suggested to be accessed first is placed above (refer to the Sect. 4).
5. The micro OER, which is more related to the learners' education background, or falls in the relevant disciplines or inter-disciplines is placed with priority if the disciplinary difference between this two candidate micro OERs is obvious.
6. Otherwise the recommended micro OER list is randomly ordered if none of the above rules applies.

4.2 Learning Path Establishment

Candidate learning path solutions (chromosomes) are randomly generated where each of them is a learning path with a series of micro OERs, rather than an individual micro OER. For a chromosome, its violation degree is investigated by examining the relations between each contiguously prior/posterior micro OER pair against the first 5 rules listed in the previous Sect. 4.1, and then summing up. For such pair in a chromosome, its violation degree, $VD(MR^t, MR^{t+1})$, is calculated by the weighted sum of its violations against rule 2 to rule 5, respectively, where MR^t is the t^{th} micro OER in k and MR^{t+1} is the $t + 1^{th}$. The higher the violation degree is, the more serious the candidate's learning path violates the rules. The violation degree of a candidate learning path, k, is calculated using the following Eq. (1)

$$VD_k = \sum VD(MR^{\mathrm{t}}, MR^{t+1}) \tag{1}$$

A given micro learning resource MR_u's real-time suitability for micro learning, $RT_{u,\ j}$, is calculated in a previous work [9]. We borrow the definition of this variable. Hence, for the candidate learning path, k, $RT_{k,j}$ denotes the sum of the real-time suitability of micro OERs it contains. Similarly, $P_{k,j}$ sums up all the predicted preferences from the learner L_j versus micro OERs that k contains.

$$\eta = \min(\alpha VD_k + \beta RT_{k,j} + \gamma \big/ P^1_{k,j} + \delta / P_{k,j}) \tag{2}$$

where α, β, γ and δ serve as weight for each variable and suggestively $\alpha > \beta > \gamma > \delta$, P^1_{kj} denotes the L_k's preference value of the first micro OER in k. A heuristic algorithm in [9] infers a few optimized learning paths, by minimizing the fitness of each. A recommendation list is generated with a size of N, namely N learning paths with lowest finesses are selected and their first micro OERs are placed in the recommendation list. Once the learner makes his/her first option micro open learning, the following items in the same learning path will be adapted to him/her sequentially.

4.3 Optimization of Micro OER Similarity Calculation

Hereby, the similarity calculation among micro OERs is crucial to the quality of item-based collaborative filtering approach. Using the Eq. (3), this calculation is not only based on their Euclidean distance on educational settings, but also added a time decay factor, which considers accumulation and attenuation of interest, and a penalty term, which tackles the filter bubbles. These two operators are shown as the latter multipliers in Eq. (3).

$$sim^M(n,g) = \sum (|n,g|) * N_0 e^{-\lambda(t_1 - t_2)} * \frac{1}{\log_a(O_j + c)} \tag{3}$$

where the t_1 is the current time and t_2 is the time when the existing micro OER, MR_g, was accessed. O_j refers to the times of a specific learner, L_j's operation, retrieved from the real-time MLaaS usage (as stated in Sect. 4). The constant c keeps the denominator unequal to zero.

The L_j's preference values are selected and evaluated to obtain their mean. A K nearest neighbor (KNN) algorithm is able to cluster items with higher similarities with the new micro OER, MR_n. Its neighbors form as a set, G. Consequently, the prediction for the L_j's preference against a new micro OER MR_n, is calculated by the Eq. (4):

$$P_{n,j} = \overline{P_j} + \frac{\sum_{g \in G} sim^M(MR_n, MR_g) * (P_{g,j} - \overline{P_j})}{\sum_{g \in G} sim^M(MR_n, MR_g)} \tag{4}$$

4.4 Inserting New Micro OERs into Established Learning Path

The new items will be inserted into established learning paths according to the Algorithm 1:

Algorithm 1: Insert New OER into Established Learning Path

Input: MR_n (a newly published micro OER), the set of existing micro OERs, Established learning paths, Semantic Relationships of OERs (as defined in the Section 5)
Output: Optimized learning paths which contain the MR_n inserted at the suitable position

```
begin: Investigate the $MR_n$ in terms of measurements in the Section 3.
    Measure its similarities with existing micro OERs using eq(3)
    Use KNN algorithm to cluster $MR_n$'s neighbors
    //Let G denote the set of neighbors of $MR_n$
    $MR_n$'s semantic relationships among neighbors in G are firstly examined according to the standard provided in Section 3.
    Invoke established learning paths or use the Algorithm in [9] to produce candidate learning paths
    for each $MR_g \in G$ do
        select learning paths that contain $MR_g$
        cut a segment that contain $MR_g$ and few micro OERs prior/posterior to it in each learning path
     //find rough positions where the $MR_n$ might be located at
        for each segment do
        $MR_n$'s semantic relations among micro OERs in the few places are examined again
            if there is a 'RequiredSequence'
            locate the place for $MR_n$
                interposition it between the micro OERs according to this strong order (as in the Section 3)
            end if
            if there is a 'RecommendedSequence'
                put $MR_n$ among the micro OERs according to this weak order
                or
                put $MR_n$ in parallel with one of existing micro OERs alternatively
                compare the predicted preference values of $MR_n$ and all existing micro OERs' in this segment
                apply the rule 2 in the Section 4.1
            measure the fitness of new learning paths using eq(2)
            use Hill Climbing algorithm to compare fitness
            replace an existing OER in established learning path with worse fitness
            or
            insert $MR_n$ between two existing micro OERs, keep $MR_n$ added on as extra if the fitness is satisfactory //in this case the overall quantity of items in the new learning path is increased
            end if
        end for
    end for
generate new learning paths containing $MR_n$
end
```

A Hill Climbing algorithm effectively abandons the learning paths with poor fitness by searching locally and reduces the times of iterations [14]. This approach does not examine throughout all elements in all matrices, hence its computing complexities and running time are acceptable for online computation which is in demand of fast response.

5 Evaluation

In this section we evaluate the qualities of generated learning path with newly published micro OERs inserted. We borrow the concept of 10-cross validation, by dividing the micro OERs in the relevant fields into two portions, in a ratio of 1:9. Learning paths were generated among the nine-tenth micro OERs. For each learning path, one micro OER from the rest is selected and treated as a newly published micro OER in the experiment. In total, 3674 micro OERs in the information technology field, 4479 micro OERs in the business field and 3254 in the social science field are picked up as candidates; and 366, 448 and 325 of them were selected out as test items, respectively. The Algorithm 1 is executed to find a place for each new member to the majority.

The Fig. 2 gives the violation degree (i.e. *VD* as defined in the Sect. 4.2) for the learning paths with new micro OERs inserted generated by using MLaaS approach against those generated by using the shortest-path approach [15] and competency-based approach [16]. In the information technology field 2044 new learning paths come out, while in the business and social science fields the numbers are 3746 and 2329. This is because one or more places are found for a new micro OERs; or according to the Algorithm 1, two new learning paths are generated when there is a 'RecommendedSequence' relation. The shortest-path (SP) approach and competency-based (C-based) approach are executed as well to put newly published micro OERs into places among or in parallel with items in established learning paths. Actually, the working principles of SP and C-based are not finding a place for the newly published micro OER within the established learning path, while rebuilding a new learning path thoroughly. According to the Fig. 2, the average violation degrees of the learning paths generated separately by the three approaches are compared in terms of the three disciplines. It can be found that MLaaS approach outperforms SP and C-based approach overall, as average *VD*s of MLaaS-generated learning paths in each discipline are far less than the others. Also, the SP approach is difficult to identify a reasonably learning path provided that there are many micro OERs loosely correlated (i.e. with the weak order RecommendedSequence).

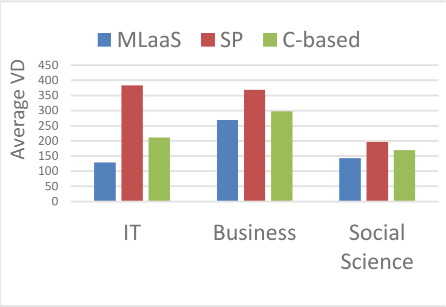


Fig. 2. Performance comparison – average VDs of all generated learning paths

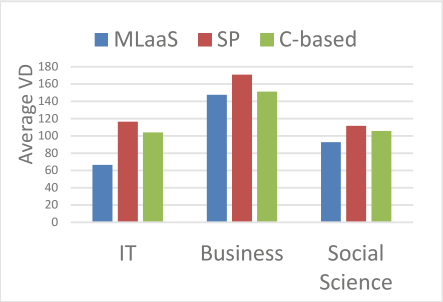


Fig. 3. Performance comparison – average VDs of the best learning paths for each test micro OER

Afterwards, we evaluate the best solutions that are produced by MLaaS, SP and C-based. Therein, one test micro OER is only allowed to be involved in one learning path; and for each test object, the new learning path with lowest *VD* value is selected. This is to eliminate the potential influences brought by the loose coupling of prior/posterior micro OER pairs. In this case, the amount of the nominated learning paths exactly equals to the amount of test micro OERs, namely 366, 448 and 325, respectively.

The results in the Fig. 3 show same observation that the MLaaS approach surpasses the other two approaches, by finding better places to insert newly published micro OERs meanwhile breaking the rules (as described in Sect. 3) less times. It is worth noting that the average violation difference is considerably larger in the information technology discipline. This is probably because the learning paths in this field are relatively longer. In other words, an individual learning path that goes through a complete information technology knowledge area consists of more micro OERs than those of business and social science. In addition, all of the three approaches produce learning paths with higher violation degrees in the business discipline. It can be potentially attributed to that micro OERs having other OERs as knowledge pre-requirement, and one micro OER being closely related to more than one of the others, are more often found in the business discipline.

6 Conclusion

In this paper we have introduced a heuristic approach to overcome the new-item cold start problem in micro OER recommendation. The newly published micro OERs are evaluated using an optimized similarity, by considering the accumulation and attenuation of interest and filter bubbles. Experiment results have proved that the proposed approach can generate learning paths with higher conformity with heuristic rules, hence finding more appropriate places in established learning paths for new items.

Acknowledgement. This research has been conducted with the support of the Australian Research Council Discovery Project, DP180101051.

References

1. Hylen, J., Van Damme, D., Mulder, F., D' Antoni, S.: Open educational resources: analysis of responses to the OECD country questionnaire. OECD Education Working Papers No. 76, June 2012
2. Souza, M.I., Amaral, S.F.D.: Educational micro content for mobile learning virtual environments. Creat. Educ. **5**, 672–681 (2014)
3. Kovachev, D., Cao, Y., Klamma, R., Jarke, M.: Learn-as-you-go: new ways of cloud-based micro-learning for the mobile web. In: Leung, H., Popescu, E., Cao, Y., Lau, R.W.H., Nejdl, W. (eds.) ICWL 2011. LNCS, vol. 7048, pp. 51–61. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-25813-8_6
4. Bruck, P.A., Motiwalla, L., Foerster, K.: Mobile learning with micro-content: a framework and evaluation. In: 25th Bled eConference, Bled, Slovenia, pp. 527–542 (2012)

5. Sun, Z., Guo, G., Zhang, J.: Exploiting implicit item relationships for recommender systems. In: Ricci, F., Bontcheva, K., Conlan, O., Lawless, S. (eds.) UMAP 2015. LNCS, vol. 9146, pp. 252–264. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-20267-9_21
6. Sun, G., Cui, T., Beydoun, G., Chen, S., Dong, F., Xu, D., Shen, J.: Towards massive data and sparse data in adaptive micro open educational resource recommendation: a study on semantic knowledge base construction and cold start problem. Sustainability **9**(6), 898.1–898.21 (2017)
7. Sun, G., Cui, T., Guo, W., Beydoun, G., Xu, D., Shen, J.: Micro learning adaptation in MOOC: a software as a service and a personalized learner model. In: Li, F.W.B., Klamma, R., Laanpere, M., Zhang, J., Manjón, B.F., Lau, Rynson W.H. (eds.) ICWL 2015. LNCS, vol. 9412, pp. 174–184. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-25515-6_16
8. Sun, G., Cui, T., Shen, J., Xu, D., Beydoun, G., Chen, S.: Ontological learner profile identification for cold start problem in micro learning resources delivery. In: The 17th IEEE International Conference on Advanced Learning Technologies (ICALT), Timisoara, Romania, pp. 16–20 (2017)
9. Sun, G., Cui, T., Xu, D., Chen, H., Chen, S., Shen, J.: Assisting open education resource providers and instructors to deal with cold start problem in adaptive micro learning: a service oriented solution. In: The 14th IEEE International Conference on Services Computing (SCC), Hawaii, USA, June 2017, pp. 196–203 (2017)
10. Lika, B., Kolomvatsos, K., Hadjiefthymiades, S.: Facing the cold start problem in recommender systems. Expert Syst. Appl. **41**(4), 2065–2073 (2014)
11. Miranda, S., Mangione, G.R., Orciuoli, F., Gaeta, M., Loia, V.: Automatic generation of assessment objects and remedial works for MOOCs'. In: 12th International Conference on Information Technology Based Higher Education and Training, Antalya, Turkey, October 2013
12. Nawrot, I., Doucet, A.: Building engagement for MOOC students', introducing support for time management on online learning platforms. In: Proceeding of WWW 2014 Companion (2014)
13. Moreno, A., Valls, A., Isern, D., Marin, L., Borras, J.: SigTur/E-Destination: ontology-based personalized recommendation of tourism and leisure activities. Eng. Appl. Artif. Intell. **26** (1), 633–651 (2013)
14. Tam, V., Lam, E.Y., Fung, S.T.: A new framework of concept clustering and learning path optimization to develop the next-generation e-Learning systems. J. Comput. Educ. **1**(4), 335–352 (2014)
15. Alian, M., Jabri, R.: A shortest adaptive learning path in eLearning systems: mathematical view. J. Am. Sci. **5**(6), 32–42 (2009)
16. Hsu, W.-C., Li, C.-H.: A competency-based guided-learning algorithm applied on adaptively guiding e-Learning. Interact. Learn. Environ. **23**(1), 106–125 (2015)

How Do Different Levels of AU4 Impact Metacognitive Monitoring During Learning with Intelligent Tutoring Systems?

Michelle Taub(✉), Roger Azevedo, and Nicholas V. Mudrick

Department of Psychology, North Carolina State University, Raleigh, NC, USA
{mtaub,razeved,nvmudric}@ncsu.edu

Abstract. We investigated how college students' ($n = 40$) different levels of action unit 4 (AU4: brow lowerer), metacognitive monitoring process use and pre-test score were associated with metacognitive monitoring accuracy during learning with a hypermedia-based ITS. Results revealed that participants with high pre-test scores had the highest accuracy scores with low levels of AU4 and use of more metacognitive monitoring processes, whereas participants with low pre-test scores had higher accuracy scores with high levels of AU4 and use of more metacognitive monitoring processes. Implications include designing adaptive ITSs that provide different types of scaffolding based on levels of prior knowledge, use of metacognitive monitoring processes, and emotional expressivity keeping in mind that levels of emotions change over time, and therefore must be monitored to provide effective scaffolding during learning.

Keywords: Affective and metacognitive processes · Hypermedia-based ITS
Process data · Self-regulated learning

1 Introduction

Self-regulated learning (SRL) implies students play an active role during learning, as opposed to being passive recipients of information and involves the use of cognitive, affective, metacognitive, and motivational (CAMM) processes [1]. However, students do not typically deploy effective CAMM SRL processes during learning [1], and as such, researchers have designed ITSs, which focus on fostering specific CAMM processes. Limited research has investigated participants' use of multiple CAMM processes (e.g., emotions with metacognition) during learning with ITSs. Therefore, for this study, we examined how students' emotions impacted the use of metacognitive processes during learning about the circulatory system with an ITS.

1.1 Previous Research

Studies have investigated the relationship between cognitive and metacognitive SRL processes with ITSs, such as metacognitive monitoring accuracy and regulatory strategies [2]. In addition, studies have investigated students' emotions during learning with

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 223–232, 2018.
https://doi.org/10.1007/978-3-319-91464-0_22

ITSs [3]. However, few studies have investigated both metacognitive and affective processes during learning with ITSs [4]. Studies have also examined how action units (AUs), which identify facial areas associated with emotions and learning outcomes (e.g., AU4 with frustration; [5]).

Research has investigated how frequency of use of SRL processes and prior knowledge impacts learning with MetaTutor, however studies have not investigated how different levels of emotions interact with metacognitive process use and prior knowledge to impact SRL, which is the goal of the current study.

1.2 Theoretical Frameworks and Current Study

We included two theoretical frameworks: (1) the Information Processing Theory (IPT) [6], as it is the only model of SRL that views SRL as an event that unfolds over time and has been used to understand cognitive and metacognitive processes during learning with ITSs, but does not include emotions; and (2) the Model of Affective Dynamics [7], as it focuses exclusively on emotions during learning with ITSs, but does not include SRL. These models are appropriate since we examined how metacognitive monitoring and emotions impacted SRL during learning with an ITS.

We used pre-test score to examine prior knowledge. We examined instances of using metacognitive monitoring processes, and defined metacognitive monitoring process use as the order the process was used (i.e., a score of 4 means it is the 4th instance). To examine emotions, we assessed the evidence score of AU4 (brow lowerer) during each instance of metacognitive monitoring, where evidence score is defined as the likelihood of a human coding for the presence or absence of the AU. We examined the correctness of each metacognitive process using a correctness score ratio.

Our research questions were: (RQ1): Is there a relationship between metacognitive monitoring process order number and correctness score ratio of metacognitive monitoring processes, and does this relationship depend on pre-test ratio?; (RQ2): Is there a relationship between AU4 evidence score and correctness score ratio, and does this relationship depend on pre-test ratio?; and (RQ3): Does the relationship between metacognitive monitoring process number and correctness score ratio depend on AU4 evidence score and pre-test ratio?

We hypothesized (H1): A significant interaction: participants with the highest correctness score ratio would use more metacognitive monitoring processes and have higher pre-test ratios; (H2): A significant interaction: participants with the highest correctness score ratio will have higher evidence scores of AU4 and high pre-test ratios; and (H3): There will be a significant three-way interaction: participants with the highest correctness score ratio will use more metacognitive monitoring processes, have high AU4 evidence scores, and high pre-test ratios.

2 Methods

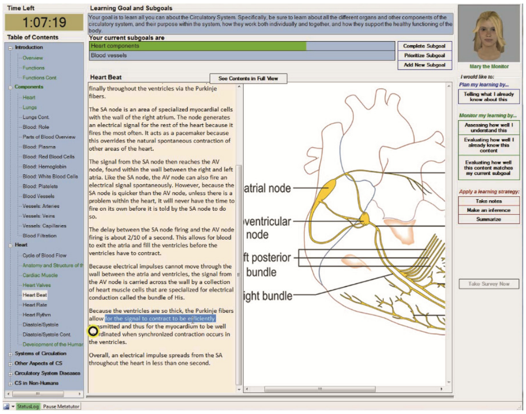
2.1 Participants and Materials

40[1] undergraduate students (55% female) from a North American university participated (M_{age} = 19.8, SD_{age} = 2.13) in this study. They were compensated $10/hour.

We administered pre- and post-tests, and self-report questionnaires on emotions and motivation. The tests were 30-item, multiple-choice tests on the circulatory system. Pre-test scores ranged from 7 (23%) to 23 (77%), M = 17.3 (58%), SD = 4.22.

2.2 MetaTutor

MetaTutor is a hypermedia-based ITS that teaches participants about the circulatory system [1] while supporting their use of cognitive and metacognitive SRL processes. The environment contains 47 pages of text and static diagrams, where participants could navigate to accomplish their overall learning goal of learning as much as they could about the human circulatory system. The interface (Fig. 1) was designed to foster the effective use of planning, monitoring, and strategizing [1].



Your goal is to learn all you can about the Circulatory System. Specifically, be sure to learn about all the different organs and other components of the circulatory system, and their purpose within the system, how they work both individually and together, and how they support the healthy functioning of the body.

finally throughout the ventricles via the Purkinje fibers.

The SA node is an area of specialized myocardial cells with the wall of the right atrium. The node generates an electrical signal for the rest of the heart because it fires the most often. It acts as a pacemaker because this overrides the natural spontaneous contraction of other areas of the heart.

The signal from the SA node then reaches the AV node, found within the wall between the right and left atria. Like the SA node, the AV node can also fire an electrical signal spontaneously. However, because the SA node is quicker than the AV node, unless there is a problem within the heart, it will never have the time to fire on its own before it is told by the SA node to do so.

The delay between the SA node firing and the AV node firing is about 2/10 of a second. This allows for blood to exit the atria and fill the ventricles before the ventricles have to contract.

Because electrical impulses cannot move through the wall between the atria and ventricles, the signal from the AV node is carried across the wall by a collection of heart muscle cells that are specialized for electrical conduction called the bundle of His.

Because the ventricles are so thick, the Purkinje fibers allow for the signal to contract to be efficiently transmitted and thus for the myocardium to be well coordinated when synchronized contraction occurs in the ventricles.

Overall, an electrical impulse spreads from the SA throughout the heart in less than one second.

atrial node
ventricular node
ft posterior bundle
ight bundle
Mary the Monitor
I would like to:
Plan my learning by...
Telling what I already know about this
Monitor my learning by...
Assessing how well I understand this
Evaluating how well I already know this content
Evaluating how well this content matches my current subgoal
Apply a learning strategy:
Take notes
Make an inference
Summarize
Take Survey Now

Fig. 1. Screenshot of the MetaTutor interface.

MetaTutor has four pedagogical agents (PAs), who are each responsible for one aspect of SRL. Gavin the Guide administers questionnaires. Pam the Planner assists with setting sub-goals. Sam the Strategizer helps participants use cognitive strategies (e.g., summarizing). Mary the Monitor focuses on metacognitive monitoring, and assists participants with judging how well they understand the content (judgment of learning JOL), assessing if they had already seen content before (feeling of knowing; FOK), evaluating the relevancy of the current page and image to their current sub-goal (content evaluation, CE), and monitoring if they read sufficient information to complete their

[1] This is a subset of a sample of 62 participants, as we did not include participants who did not have facial expression data.

sub-goal (monitoring progress towards goals; MPTG). One PA is present at a time, based on the activities participants are engaging in, and the level of involvement depends on the assigned experimental condition (see below).

2.3 Experimental Procedure

MetaTutor is a 2-day study, where on day 1, participants completed a consent form, demographics questionnaire, self-report questionnaires, and the pre-test. On day 2, participants learned with MetaTutor. First, the equipment was calibrated. Next, they viewed introductory videos about how to use the system and use SRL processes. Participants then completed the sub-goal setting phase, followed by the 90-min learning session. Once the learning session ended, participants completed the post-test and questionnaires, were debriefed, thanked for participating, and paid for their time.

Participants were randomly assigned to one of two conditions. In the *prompt and feedback* condition, the PAs prompted participants to engage in SRL processes, and provided feedback on their performance. In the *control* condition, the PAs did not provide any prompts or feedback.

2.4 Coding and Scoring

We collected multi-channel process data during learning, including (1) log files, which captured input into the system (in ms); and (2) video recordings that were run through facial recognition software to determine the emotions participants expressed.

Log files included each instance of metacognitive monitoring process use. We examined each instance of when participants used JOLs, FOKs, CEs, and MPTGs (see MetaTutor section), and assigned a correctness score ratio to each instance, resulting in each participant having multiple rows of data, depending on the number of metacognitive monitoring processes ($M = 14$, $SD = 9.54$). Correctness score ratio was calculated based on each process. Specifically, page quizzes (taken after JOLs and FOKs) included three multiple choice questions, and were scored using weighted correctness (out of 3), sub-goal quiz scores (i.e., MPTGs) included 10 multiple-choice questions, and were scored using weighted correctness (out of 10), and CEs were scored based on the correctness of participants' relevancy judgments. For example, if both text and diagram were relevant and they answered 'both', they received a score of 1 (0.5 for knowing it was relevant, and 0.5 for naming both relevant items), however if they answered 'page only', they received a score of 0.75 (0.5 for knowing it was relevant, but only 0.25 because they only named one relevant item). We used pre-test score ratio to examine how pre-test related to the use of metacognitive monitoring processes and levels of emotions during learning.

We used Attention Tool 6.1 to obtain evidence scores for specific action units, which are designated areas on the face that contribute to facial expressions of different emotions (e.g., eyebrow lowerer). Evidence scores are values that indicate the likelihood of an emotion or action unit being present or absent as would be coded by a human coder, which increases exponentially. An evidence score of 1 indicates that 10 human coders are likely to code for that emotion or action unit, a score of 2 indicates a likelihood of

100 coders, etc. As the data is collected at a frequency of 30 Hz, we averaged evidence scores for the duration of each metacognitive monitoring process.

Preliminary analyses using four learning-related AUs [8] indicated a significant association between AU4 (brow lowerer) and correctness score ratio ($p < .05$), however AU5 (upper lid raiser), AU14 (dimpler), and AU15 (lip corner depressor) were not significant predictors of correctness score ratio. Thus, for subsequent analyses, we only included evidence score for AU4 (brow lowerer) (Fig. 2).

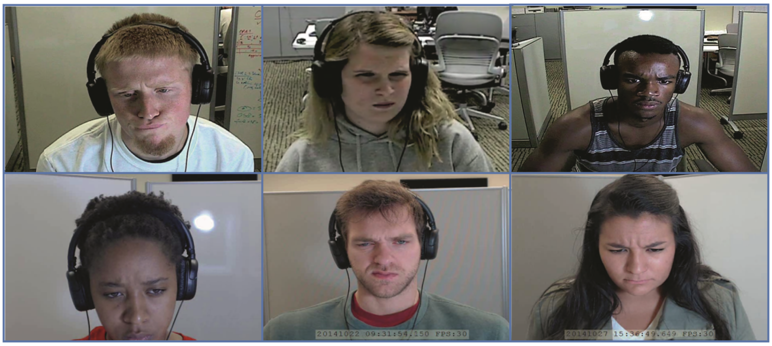


Fig. 2. Participants expressing AU4 while using MetaTutor.

3 Results

For this study, we used multi-level modeling (MLM). We did not include experimental condition or session duration because preliminary analyses revealed they were not significant predictors ($p > .05$). A fully unconditional model (no predictor variables) revealed significant between- ($\tau_{00} = .012$, $z = 2.38$, $p = .0086$) and within-subjects ($\sigma^2 = .098$, $z = 16.28$, $p < .0001$) variance in correctness score ratio. The intraclass correlation coefficient (ICC) revealed 11.1% of the variance was between- and 88.9% of the variance was within-subjects.

3.1 Is There a Relationship Between Metacognitive Monitoring Process Use and Correctness Score Ratio of Metacognitive Monitoring Processes and Does this Relationship Depend on Pre-test Ratio?

We ran a non-randomly varying slopes model with metacognitive monitoring process use and pre-test ratio as the level 1 and 2 predictors, respectively, and correctness score ratio as the dependent variable. This model used the following equations, where i = the individual, and m = metacognitive monitoring:

Level 1:

$$\mathrm{ScoreRatio}_{\mathrm{im}} = \beta_{0\mathrm{im}} + \beta_{1\mathrm{im}}(\mathrm{MMUse}) + \mathrm{r}_{\mathrm{im}} \tag{1}$$

Level 2:

$$\beta_{0i} = \gamma_{00} + \gamma_{01}(\text{PreRatio}) + u_{0i} \tag{2}$$

$$\beta_{1i} = \gamma_{10} + \gamma_{11}(\text{PreRatio}) \tag{3}$$

Results revealed an increase in metacognitive monitoring process use was associated with an increase in correctness score ratio; $\gamma_{10} = .027$, $t = 2.32$, $p = .021$; an increase in pre-test ratio was associated with an increase in correctness score ratio; $\gamma_{01} = .94$, $t = 4.72$, $p < .0001$; and a significant cross-level interaction, such that participants with the highest correctness score ratios had high numbers of metacognitive monitoring process use, but low pre-test ratios; $\gamma_{11} = -.049$, $t = -2.40$, $p = .017$ (see Fig. 3). This model accounted for 54.88% of the between-subjects variance and .62% of the within-subjects variance in correctness score ratio.

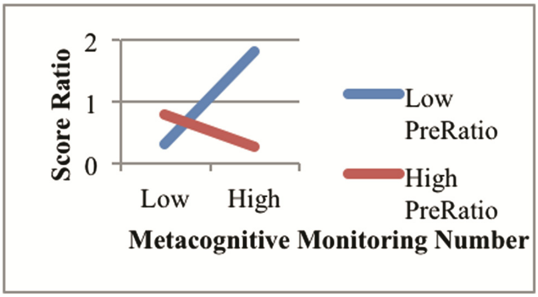


Fig. 3. Cross-level interaction with metacognitive monitoring processes and pre-test ratio.

3.2 Is There a Relationship Between AU4 Evidence Score and Correctness Score Ratio, and Does this Relationship Depend on Pre-test Ratio?

We ran a non-randomly varying slopes model with AU4 evidence score as the level 1 (within-subjects) predictor, pre-test ratio as the level 2 (between-subjects) predictor, and correctness score ratio as the dependent variable. This model used the following equations, where i = the individual, and m = metacognitive monitoring:

Level 1:

$$\text{ScoreRatio}_{im} = \beta_{0im} + \beta_{1im}(\text{AU4Evidence}) + r_{im} \tag{4}$$

Level 2:

$$\beta_{0i} = \gamma_{00} + \gamma_{01}(\text{PreRatio}) + u_{0i} \tag{5}$$

$$\beta_{1i} = \gamma_{10} + \gamma_{11}(\text{PreRatio}) \tag{6}$$

Results revealed an increase in AU4 evidence score was associated with an increase in correctness score ratio ($\gamma_{10} = .37$, $t = 1.95$, $p = .05$), an increase in pre-test ratio was associated with an increase in correctness score ratio ($\gamma_{01} = .64$, $t = 4.77$, $p < .0001$), and a significant cross-level interaction ($\gamma_{11} = -.84$, $t = -2.50$, $p = .013$). Specifically

(Fig. 4), participants with high pre-test ratios performed better on metacognitive monitoring processes with low evidence scores of AU4 (i.e., lower evidence of eyebrow lowering), however participants with low pre-test ratios had higher metacognitive monitoring correctness scores with high evidence scores of AU4. This model accounted for 57% and 1.4% of the between- and within-subjects variance in correctness score ratio, respectively. In sum, this means that accuracy of using metacognitive monitoring processes was highest for students expressing low levels of AU4 when they had high prior knowledge, however the opposite was the case for low prior knowledge, where accuracy in metacognitive monitoring processes was highest when experiencing high levels of AU4.

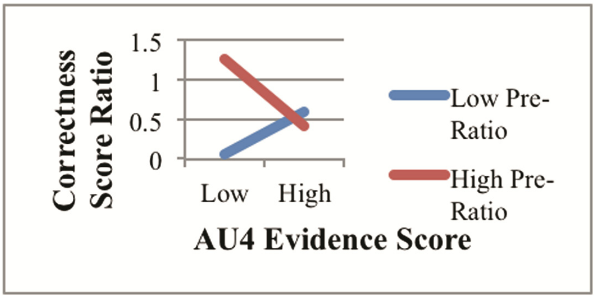


Fig. 4. Cross-level interaction with AU4 evidence score and pre-test ratio

3.3 Does the Relationship Between Metacognitive Monitoring Process Use and Correctness Score Ratio Depend on AU4 Evidence Score and Pre-test Ratio?

We ran a three-way cross level interaction model with number of metacognitive monitoring process use and AU4 evidence score as the level 1 predictors, and pre-test ratio as the level 2 predictor. We used the following equations, where i = the individual, and m = metacognitive monitoring:

Level 1:

$$\text{ScoreRatio}_{\text{im}} = \beta_{0\text{im}} + \beta_{1\text{im}}(\text{MMUse}) + \beta_{2\text{im}}(\text{AU4EvidenceScore}) + \beta_{3\text{im}}(\text{MMNumber} * \text{AU4EvidenceScore}) + r_{\text{im}} \tag{7}$$

Level 2:

$$\beta_{0\text{i}} = \gamma_{00} + \gamma_{01}(\text{PreRatio}) + u_{0\text{i}} \tag{8}$$

$$\beta_{1\text{i}} = \gamma_{10} + \gamma_{11}(\text{PreRatio}) \tag{9}$$

$$\beta_{2\text{i}} = \gamma_{20} + \gamma_{21}(\text{PreRatio}) \tag{10}$$

$$\beta_{3\text{i}} = \gamma_{30} + \gamma_{31}(\text{PreRatio}) \tag{11}$$

Results revealed no significant association between monitoring process use and correctness score ratio ($\gamma_{10} = .0002$, $t = .14$, $p = .89$), no significant association between AU4 evidence score and correctness score ratio ($\gamma_{20} = -.038$, $t = -.54$, $p = .59$), however there was a significant association between pre-test ratio and correctness score ratio ($\gamma_{01} = .66$, $t = 4.83$, $p < .0001$). Additionally, results indicated a significant interaction between metacognitive monitoring process use and AU4 evidence score ($\gamma_{30} = .049$, $t = 2.79$, $p = .0055$), and a significant three-way cross-level interaction ($\gamma_{31} = -.097$, $t = -3.06$, $p = .0023$). Specifically, participants with the lowest correctness score ratios had low pre-test ratios, high metacognitive monitoring process use and low AU4 evidence scores, compared to participants with the highest correctness score ratios who had high pre-test ratios, high uses of metacognitive monitoring processes, and low AU4 evidence scores (Fig. 5). This model accounted for 54% and 1.9% of the between-and within-subjects variance in correctness score ratio, respectively.

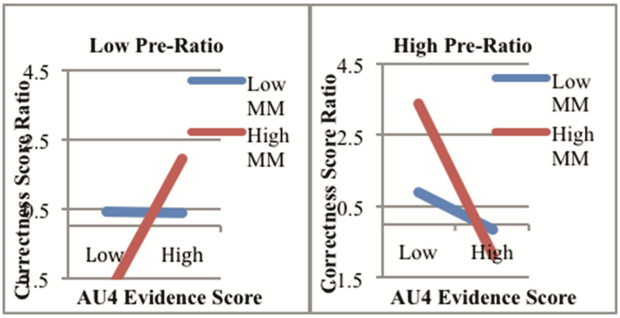


Fig. 5. Three-way cross-level interaction on correctness score ratio.

4 Discussion

Results from our study revealed participants with high pre-test ratios had the highest correctness score ratios when they used more metacognitive monitoring processes with low evidence scores of AU4 (Fig. 5, right), however participants with low pre-test ratios had the highest correctness ratios if they used more metacognitive monitoring processes, but high evidence scores of AU4 (Fig. 5, left). This suggests that emotions can impact students differently based on prior knowledge and metacognition.

Our first research question revealed participants with low pre-test ratios and use of many metacognitive processes had the highest correctness score ratios. This partially supports H1 as we predicted higher correctness ratios to be associated with high numbers of metacognitive processes and high pre-test ratios. Research question 2 revealed that the highest correctness score ratio was associated with high pre-test ratios and low evidence scores of AU4. This partially supports H2 as we predicted higher correctness ratios to be associated with high pre-test ratios, but low levels of AU4. Research question 3 revealed when combining all variables, the highest correctness ratios were for participants with high pre-test ratios, high numbers of metacognitive processes, and low levels of AU4. This partially supports H3 as we predicted high levels of pre-test ratios and metacognitive processes, however we also predicted high levels of AU4. These results demonstrate the importance of investigating variables simultaneously, which is more representative of learning as these factors play a role together.

4.1 Designing Intelligent Tutoring Systems

Our findings not only demonstrate how emotions (i.e., AU4) impacted the accuracy in metacognitive monitoring processes, but also how high vs. low evidence scores of AU4 and pre-test ratios impacted metacognitive monitoring accuracy. Therefore, these findings indicate the importance of investigating not only the presence of emotions, but also levels of experiencing them. Future studies should investigate how levels of emotions impact learning so we can determine how different levels of AUs signify different emotions. For example, what does low vs. high AU4 mean? Are different levels of these AUs indicative of different emotions, or different intensities of the same emotion? Are students with high prior knowledge experiencing low levels of confusion, or low levels of mental effort [5], which they do not need to exert because they already know the content, or are these low levels of AU4 indicative of emotion regulation? In contrast, are high levels of AU4 for low prior knowledge students indicative of high levels of confusion, or do other emotions play a role? Future studies are needed to investigate levels of AUs to determine how these different levels are interacting with other variables to impact student performance in different ways.

These results lead the way for the design of ITSs that are adaptive based on students' emotions, use of metacognitive processes, and prior knowledge, using affective computing [9]. ITSs should also be designed to be adaptive to processes in addition to affect, for example cognitive, metacognitive, and motivational processes. During learning with MetaTutor, participants can engage in cognitive (e.g., taking notes, creating summaries) and metacognitive processes (JOLs, FOKs, CEs, and MPTGs) by clicking on the SRL palette as they read. Based on their performance on these processes, their evidence scores of influencing AUs or emotions and their levels of prior knowledge, the ITS can provide them with the appropriate feedback.

When designing these ITSs, we must also keep in mind that different types of variables do or do not change over time. Specifically, if an ITS is adaptive based on levels of prior knowledge, this score does not change, however the number of metacognitive monitoring processes used will change, as can emotions. For example, at the beginning of the learning session when participants with low prior knowledge have used fewer metacognitive processes, low levels of AU4 will be more advantageous, however later in the session, when participants have used more metacognitive processes, the ITS should explain that high levels of confusion can be beneficial, and can help provide strategies on how to resolve that confusion. In contrast, for participants with high prior knowledge, different types of scaffolding might be provided at different times during the learning session, such that early in the session, they might benefit from high levels of confusion. Therefore, research needs to continue investigating the different impacts on learning with ITSs, and the levels of these variables. We can develop more ITSs that are adaptive to the many student characteristics that have been found to play an integral role in learning. Thus, we can ensure that all students are learning the most effectively and efficiently with these environments.

Acknowledgments. This research was supported by funding from the National Science Foundation (DRL#1431552; DRL#1660878, DRL#1661202) and the Social Sciences and Humanities Research Council of Canada (SSHRC 895-2011-1006). The authors would like to thank the members from the SMART Lab at NCSU for their assistance with data collection.

References

1. Azevedo, R., Taub, M., Mudrick, N.V.: Understanding and reasoning about real-time cognitive, affective, and metacognitive processes to foster self-regulation with advanced learning technologies. In: Alexander, P.A., Schunk, D.H., Greene, J.A. (eds.) Handbook of Self-regulation of Learning and Performance, 2nd edn, pp. 254–270. Routledge, New York (2018)
2. Taub, M., Mudrick, N.V., Azevedo, R., Millar, G.C., Rowe, J., Lester, J.: Using multi-channel data with multi-level modeling to assess in-game performance during gameplay with Crystal Island. Comput. Hum. Behav. **76**, 641–655 (2017)
3. D'Mello, S., Lehman, B., Sullins, J., Daigle, R., Combs, R., Vogt, K., Perkins, L., Graesser, A.: A time for emoting: when affect-sensitivity is and isn't effective at promoting deep learning. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6094, pp. 245–254. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13388-6_29
4. Chauncey Strain, A., Azevedo, R., D'Mello, S.: Exploring relationships between learners' affective states, metacognitive processes, and learning outcomes. In: Cerri, S.A., Clancey, W.J., Papadourakis, G., Panourgia, K. (eds.) ITS 2012. LNCS (LNAI; LNB), vol. 7315, pp. 59–64. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-30950-2_8
5. Grafsgaard, J.F., Wiggins, J.B., Boyer, K.E., Wiebe, E.N., Lester, J.C.: Automatically recognizing facial expression: predicting engagement and frustration. In: 6th International Conference Educational Data Mining, EDM 2013, pp. 43–50 (2013)
6. Winne, P.H.: Cognition and metacognition within self-regulated learning. In: Alexander, P.A., Schunk, D.H., Greene, J.A. (eds.) Handbook of Self-regulation of Learning and Performance, 2nd edn, pp. 36–48. Routledge, New York (2018)
7. D'Mello, S., Graesser, A.: Dynamics of affective states during complex learning. Learn. Instr. **22**, 145–157 (2012)
8. D'Mello, S.K., Craig, S.D., Graesser, A.C.: Multi-method assessment of affective experience and expressin during deep learning. Int. J. Learn. Technol. **4**, 165–187 (2009)
9. D'Mello, S., Kappas, A., Gratch, J.: The affective computing approach to affect measurement. Emot. Rev. 1–10 (2017)

How Are Students' Emotions Associated with the Accuracy of Their Note Taking and Summarizing During Learning with ITSs?

Michelle Taub[1](✉), Nicholas V. Mudrick[1], Ramkumar Rajendran[2], Yi Dong[2], Gautam Biswas[2], and Roger Azevedo[1]

[1] North Carolina State University, Raleigh, NC, USA
{mtaub,nvmudric,razeved}@ncsu.edu
[2] Vanderbilt University, Nashville, TN, USA
{ramkumar.rajendran,yi.dong,gautam.biswas}@vanderbilt.edu

Abstract. The goal of this study was to examine 38 undergraduate and graduate students' note taking and summarizing, and the relationship between emotions, the accuracy of those notes and summaries, and proportional learning gain, during learning with MetaTutor, an ITS that fosters self-regulated learning while learning complex science topics. Results revealed that students expressed both positive (i.e., joy, surprise) and negative (i.e., confusion, frustration, anger, and contempt) emotions during note taking and summarizing, and that these emotions correlated with each other, as well as with proportional learning gain and accuracy of their notes and summaries. Specifically, contempt during note taking was positively correlated with proportional learning gain; note taking accuracy was negatively correlated with proportional learning gain; and confusion during summarizing was positively correlated with summary accuracy. These results reveal the importance of investigating specific self-regulated learning processes, such as taking notes or making summaries, with future research aimed at examining the differences and similarities between different cognitive and metacognitive processes and how they interact with different emotions similarly or differently during learning. Implications of these findings move us toward developing adaptive ITSs that foster self-regulated science learning, with specific scaffolding based on each individual student's learning needs.

Keywords: Cognitive learning strategies · Facial expressions of emotion
Latent semantic analysis · Process data · Self-regulated learning

1 Introduction

Self-regulated learning (SRL) involves the monitoring and control of cognitive, affective, metacognitive, and motivational (CAMM) processes [1]. Research has indicated that although these processes can enhance learning, students often fail to deploy SRL processes effectively and efficiently. Therefore, to foster effective SRL, researchers have developed ITSs that scaffold the use of CAMM processes during learning and complex problem solving [2]. According to the information processing theory of SRL [3],

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 233–242, 2018.
https://doi.org/10.1007/978-3-319-91464-0_23

learning occurs through four cyclical phases in which information processing and SRL occur. SRL is viewed as an event that unfolds over time, and is impacted by students' cognitive, metacognitive, and motivational processes. However, the model does not focus on the affective component of SRL, thus we rely on the model of affective dynamics [4] that focuses on emotion transitions that occur when students learn with ITSs. The model is based on resolving cognitive disequilibrium that arises when students encounter an impasse (i.e., confusion), which if not resolved, can lead to frustration, and ultimately boredom (and task disengagement). We employ both models as they both address the temporal nature of CAMM SRL processes and learning with ITSs and how they change over time.

Research on ITSs has consistently demonstrated that emotions play a critical role during learning. For example, specific emotions (i.e., confusion and joy) have found to contribute to higher learning outcomes, while the presence of more negative emotions (i.e., frustration and boredom) have found to deleteriously impact learning outcomes with ITSs [1, 5]. As such, there has been a growing movement within this literature towards designing more affective-aware advanced learning technologies (AALTs) that can detect and respond to the presence of learner-centered emotions (confusion, frustration) or induce other emotions (e.g., joy) during learning [5].

Despite advancing our understanding about the influence of key learner-centered emotions on students' learning outcomes with ITSs, this area of research is still limited in certain ways. Specifically, much of this research focuses on detecting and responding to key emotions as assessed by students' self-reports. This is problematic, as self-reports cannot only interrupt students during learning, but are also subject to possible bias and error (e.g., social desirability effects). Additionally, this literature has not examined emotions during specific learning activities (e.g., summarizing and taking notes) and has instead focused on the cumulative impact of certain emotions on students' learning outcomes with these systems. As such, we address these gaps in the literature by focusing on how emotions (collected unobtrusively, by analyzing students' facial expressions) are associated with specific cognitive learning strategies (i.e., summarizing and taking notes) during learning with MetaTutor.

1.1 Current Study

The goal of the current study was to examine the relationship between students' notes and summaries, the accuracy of those notes and summaries, and proportional learning gain (PLG) with the emotions they experienced while taking those notes and creating summaries, all during learning with MetaTutor, an ITS that fosters students' SRL while learning about the human circulatory system. We focus on note taking and summarizing because they are common cognitive learning strategies students engage in and are supported by teachers, and allow students to organize and understand their thoughts as opposed to relying on memorizing the content they read [6].

We posed the following research questions: (RQ1): What is the distribution of average evidence scores of emotions during note taking and summarizing? (RQ2): What is the relationship between evidence scores of emotions, accuracy score of notes and summaries, and PLG during note taking and summarizing? (RQ3): Is there an association between evidence scores of confusion and accuracy score of summaries?

We hypothesized that (H1): There will be higher evidence scores of positive emotions and lower evidence scores of negative emotions during note taking and summarizing. (H2): Evidence scores of positive emotions expressed during note taking and summarizing will be positively correlated with accuracy score of notes and summaries and PLG. (H3): An increase in confusion will be associated with a decrease in accuracy score of summaries.

2 Methods

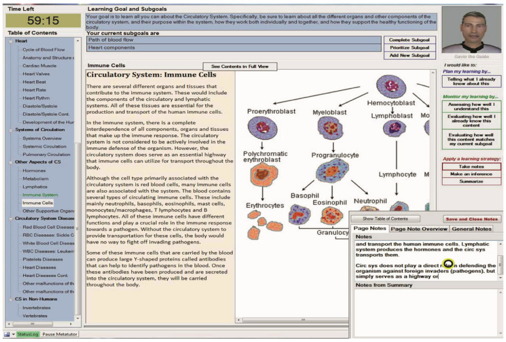
2.1 Participants and Materials

38 undergraduate and graduate students majoring in Education at a large North American university (87% female) participated ($M_{\text{age}} = 23.1$, $SD_{\text{age}} = 4.42$). Of this sample, 21 students took notes, and 19 made summaries. They were randomly assigned to 1 of 2 conditions, and were paid $10/hour for participating in the 2-day study.

Students completed a 30-item multiple choice pre-test on the circulatory system at the beginning of the study, and a counterbalanced, 30-item post test at the end of the study. They also completed self-report questionnaires on motivation, emotions, and overall system feedback at the start and end of the study.

2.2 MetaTutor: An ITS that Fosters SRL During Science Learning

MetaTutor is an ITS that fosters learning of complex science topics (circulatory system) by promoting the use of cognitive and metacognitive SRL processes during a 90-min learning session with the overall goal of learning as much as possible about the human circulatory system [1]. The system (see Fig. 1.) contains 47 pages of text and static diagrams along with other interface elements (e.g., timer, table of contents, and SRL palette) that were strategically designed to foster SRL.



Your goal is to learn all you can about the Circulatory System. Specifically, be sure to learn about all the different organs and other components of the circulatory system, and their purpose within the system, how they work both individually and together, and how they support the healthy functioning of the body.

Your current subgoals are

- Path of blood flow
- Heart components

Circulatory System: Immune Cells

There are several different organs and tissues that contribute to the immune system. These would include the components of the circulatory and lymphatic systems. All of these tissues are essential for the production and transport of the human immune cells.

In the immune system, there is a complete interdependence of all components, organs and tissues that make up the immune response. The circulatory system is not considered to be actively involved in the immune defense of the organism. However, the circulatory system does serve as an essential highway that immune cells can utilize for transport throughout the body.

Although the cell type primarily associated with the circulatory system is red blood cells, many immune cells are also associated with the system. The blood contains several types of circulating immune cells. These include mainly neutrophils, basophils, eosinophils, mast cells, monocytes/macrophages, T lymphocytes and B lymphocytes. All of these immune cells have different functions and play a crucial role in the immune response towards a pathogen. Without the circulatory system to provide transportation for these cells, the body would have no way to fight off invading pathogens.

Some of these immune cells that are carried by the blood can produce large Y-shaped proteins called antibodies that can help to identify pathogens in the blood. Once these antibodies have been produced and are secreted into the circulatory system, they will be carried throughout the body.

and transport the human immune cells. Lymphatic system produces the hormones and the circ sys transports them.

Circ sys does not play a direct role in defending the organism against foreign invaders (pathogens). but simply serves as a highway or

Fig. 1. Screenshot of the MetaTutor interface with notes open

The system contains four pedagogical agents who are each responsible for a different component of SRL. Gavin the Guide introduces the environment and administers self-report questionnaires. Pam the Planner assists with activating prior knowledge and setting and managing sub-goals. Mary the Monitor fosters metacognitive monitoring processes, such as judging one's own understanding of the content and evaluating the relevancy of the content in relation to one's current sub-goal. Sam the Strategizer focuses on cognitive learning strategies such as taking notes and summarizing. Each agent is present one at a time depending on the student's actions.

Students were randomly assigned to one of two experimental conditions prior to learning. In the *prompt and feedback* condition, students received prompts from the agents to engage in cognitive and metacognitive SRL processes, and received feedback from the agents based on their performance on these processes. In the *control* condition, the agents did not actively prompt students or provide them with any feedback. In both conditions, students could self-initiate the use of SRL processes by clicking on the SRL palette. For this study, we included note-taking or summarizing instances (user- or system-initiated) as we wanted to determine the accuracy of the notes and summaries, regardless of who prompted it.

2.3 Experimental Procedure

The study lasted 2 days. On day 1, students completed consent and demographics forms and self-report questionnaires on emotions, motivation, personality, and epistemic beliefs, followed by the 30-item pre-test.

On day 2, the experimenter began by calibrating students' video of facial expression of emotions by having them sit still in a neutral position. Next, students were shown introductory videos, which introduced the MetaTutor system as well as the importance of using SRL processes during learning. Students then set two sub-goals with Pam the Planner (i.e., sub-goal setting phase). Once they set 2 sub-goals, they began learning with MetaTutor for a total of 90 min. During learning, they could select content to read along with viewing the respective diagrams while also engaging in SRL processes, such as taking notes and making summaries. Students could engage in these processes multiple times during learning. Once the learning session was complete, students took the 30-item counterbalanced post-test, followed by more self-report questionnaires on emotions and feedback about the agents and the system itself. Students were then debriefed, paid, and thanked for participating.

2.4 Data Coding and Scoring

We collected multichannel data during learning: log files, videos of facial expressions of emotions, eye tracking, and electrodermal activity. For this study, we used log files (i.e., all student input into the system, such as quiz or test responses and content of notes and summaries typed, as well as their timestamps, at the ms-level) and videos of facial expressions (i.e., evidence scores) of emotions only.

To calculate proportional learning gain (PLG), we used the formula from [7], which used pre- ($M = 18.38$, $SD = 4.73$, range: 9–28 out of 30 for students who took notes;

M = 18.74, SD = 4.99, range: 9–28 out of 30 for students who made summaries) and post-test (M = 21.95, SD = 3.06, range: 16–29 for students who took notes; M = 21.90, SD = 3.30, range: 16–29 out of 30 for students who made summaries), accounting for the increase in correct responses from pre- to post-test.

To determine the accuracy of students' notes and summaries during learning with MetaTutor, we conducted latent semantic analysis (LSA; [8]). We used a one-to-many comparison approach, allowing us to compare notes and summaries to the topic page they used to make these notes and summaries. The semantic space that we selected was 'General reading up to first year of college' (http://lsa.colorado.edu/). The output semantic vectors were used to assess the accuracy of students' notes and summaries, with higher values indicating more semantic overlap with the target text (i.e., higher quality, where 1 = 100% overlap), and lower values indicating less semantic overlap (i.e. lower quality). Of the students who took notes and made summaries, their mean LSA scores were .68 (SD = .29) for notes, and .60 (SD = .34) for summaries.

To extract emotions, we used FACET, an empirically tested [9] automatic facial expression detection software from iMotions (https://imotions.com/facial-expressions/). Students' facial expressions recorded using a webcam were provided as input to the iMotions facial expression analyzer. The iMotions software extracted facial features to predict Action Units (AUs) [10] using a support vector machine algorithm (SVM) [11]. The facial expression software provided evidence values (defined as the log (base 10) likelihood of human coders coding for that emotion; i.e., a code of 1, 2, or 3 is the likelihood of 10, 100, or 1000 human coders coding for that emotion, respectively) associated with the students' facial expressions. This was done for six basic emotions (joy, anger, surprise, fear, disgust, and sadness) along with additional characterizations, such as confusion, frustration, contempt, and neutral. We investigated joy, anger, surprise, contempt, confusion, and frustration for our analyses because they relate closely to learning with ITSs.

The input video to the iMotions software contained 25 frames (photos) per second. The classifier analyzed these frames and predicted the value of a learner's emotions for each frame. Hence, the raw output file from iMotions contained 25 evidence score values for the 10 emotions listed above per second. We preprocessed the raw output files to aggregate emotion values per second and transformed the evidence score to a representation that could be easily used for our analysis. Our preprocessing step is briefly described below:

1. Based on our consultation with experts from iMotions, we replaced the non-zero values in the raw data to zero.
2. The standard feature rescaling process was applied to convert the evidence values of each emotion to scale the range in [0, 1]. The formula used for rescaling was:

$$rescaled(x) = \frac{x_i - \min(\mathrm{x})}{\max(x) - \min(x)} \tag{1}$$

3. To remove the noise in the data represented as a sudden spike of evidence score for an emotion, we applied a median filtering process, using a window size of 10 observations. The center evidence value in the window was replaced with the median of

all the evidence values in the window. The window size was selected after several iterations with varied window size.

4. To remove noise and to smoothen the data we applied a standard mean filter, with a window size of 25. The window size covered a one-second interval. The mean filtering replaces the center evidence value in the window with the mean score of all the evidence values in the window.
5. The evidence values of emotions per second were computed as the mean of all evidence value observations over a period of one second.

3 Results

3.1 Research Question 1: What Is the Distribution of Average Evidence Scores of Emotions During Note Taking and Summarizing?

Histograms (Fig. 2) revealed both note taking and summarizing had similar distributions, with anger scoring the highest evidence value, followed by confusion, frustration, surprise, joy, and lastly contempt. There were similar evidence scores for anger, confusion, frustration, and surprise, however there here higher evidence scores for joy and lower evidence scores for contempt during note taking compared to summarizing.

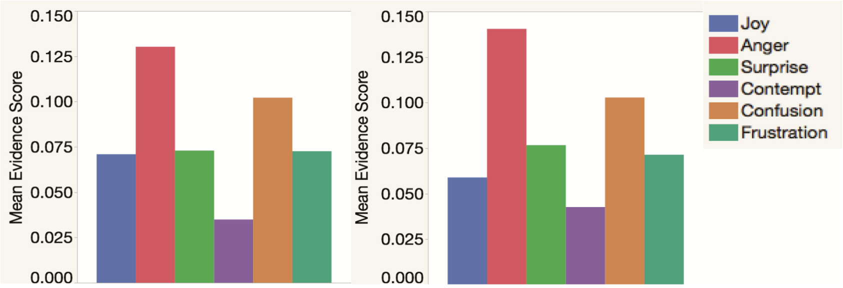


Fig. 2. Distributions of emotions during note taking (left) and summarizing (right)

3.2 Research Question 2: What Is the Relationship Between Evidence Scores of Emotions, LSA Score, and PLG During Note Taking and Summarizing?

We ran two correlations: one for note taking and one for summarizing, with PLG score, LSA score, and average evidence score for joy, anger, surprise, contempt, confusion, and frustration. Results revealed that for note taking, many emotions were correlated with each other (see Table 1). Contempt correlated positively with PLG ($r(19) = .46$, $p = .035$), and LSA score was negatively correlated with PLG ($r(19) = -.51$, $p = .018$).

For summarizing, we again found emotions to be correlated with each other (see Table 2). In addition, results revealed a significant positive correlation between confusion and LSA score ($r(17) = .47$, $p = .042$), such that higher levels of confusion were associated with higher LSA scores on summaries.

Table 1. Correlations between PLG, LSA score, and emotions for note taking

	1.	2.	3.	4.	5.	6.	7.	8.
1. PLG	–							
2. LSA score	−.51*	–						
3. Joy	.32	−.17	–					
4. Anger	−.26	−.11	.19	–				
5. Surprise	−.07	−.13	.23	.41	–			
6. Contempt	.46*	−.40	.59**	.19	−.04	–		
7. Confusion	−.01	.10	−.05	.49*	.09	.08	–	
8. Frustration	.11	−.22	.37	.71***	.09	.55*	.53*	–

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

Table 2. Correlations between PLG, LSA score, and emotions for summarizing

	1.	2.	3.	4.	5.	6.	7.	8.
1. PLG	–							
2. LSA score	−.21	–						
3. Joy	.27	.04	–					
4. Anger	−.41	.41	.15	–				
5. Surprise	−.24	.02	−.05	.25	–			
6. Contempt	.24	−.07	.67**	−.18	.07	–		
7. Confusion	.04	.47*	.11	.49*	−.16	−.22	–	
8. Frustration	.23	.19	.60**	.59**	−.08	.23	.63**	–

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

3.3 Research Question 3: Is There an Association Between Evidence Scores of Confusion and LSA Score During Summarizing?

We examined each instance of engaging in summarizing, and the evidence and LSA scores that were associated with each of those instances. We used multi-level modeling to examine the association between evidence score of confusion and LSA score (as they were significantly correlated). We did not examine PLG because our sample size was too small, thus these models did not converge.

Results from our one-way ANCOVA with random effects model did not yield a significant effect (γ_{10} = .54, t = 1.61, p = .11), revealing that confusion did not predict LSA score during summary instances.

4 Discussion and Future Directions for ITSs

The goal of this study was to investigate students' emotions during note taking and summarizing while they learned with MetaTutor, and how these emotions were associated with the accuracy of their notes and summaries, and proportional learning gain.

Results from our first research question revealed that students facially expressed a range of different emotions during note taking and summarizing, with high levels of anger, confusion, and frustration and lower levels of surprise, joy, and contempt. This did not confirm H1 because we found higher (not lower) levels of negative emotions during note taking and summarizing, which can be attributed to the fact that FACET observed more negative than positive emotions. This result contributes to research on the presence as well as benefits of negative emotions expressed during learning. For example, there is evidence that all students express these emotions during learning, and specifically during note taking and summarizing, and that these negative emotions are not necessarily detrimental to learning [12]. Additionally, these findings highlight the contribution of the role of negative emotions during strategies involved in knowledge construction activities and metacognitive monitoring, and take away from knowledge acquisition processes including reading and inspecting diagrams.

Our second research question revealed that evidence scores of emotions correlated with each other during both note taking and summarizing. Contempt was positively correlated with PLG and LSA score was negatively correlated with PLG during note taking, and confusion was positively correlated with LSA score during summarizing. This did not confirm H2 as we found positive (not negative) correlations between emotions, and between PLG and LSA score, but we did not find significant correlations between most emotions and PLG or LSA score. This negative correlation between LSA score and PLG during note taking might indicate that students were devoting too much time to their notes, which could have negatively impacted their learning outcomes because the content they were summarizing did not cover all the content on the post-test. Alternatively, the post-test themselves might not be correctly assessing all the content provided to the students to read during the learning session, leading to a mismatch between page content and post-test. Future studies should aim to match the summaries they made to the content on the post-test to investigate this further.

Our third research question revealed that confusion did not predict LSA score during summarizing, which did not confirm H3, as we predicted a significant association between these two variables. Possible statistical explanations for this result could be due to our small sample size. Lastly, it is possible that it may be that students' confusion was elicited by the summaries they were writing, and not the other way around, resulting in a non-significant model.

These findings have implications for understanding the complex nature of SRL as an event that unfolds over time, and how different SRL processes can be from each other (i.e., notes vs. using metacognitive monitoring). For example, by assessing instances of note taking and summarizing, we investigated how students used cognitive strategies during SRL and how their accuracy and levels of emotions during note taking and summarizing were associated with each other. Future studies should examine metacognitive processes and emotions exhibited while engaging in these processes, as emotions might play a different role (e.g., confusion correlating negatively with metacognitive processes as opposed to the positive correlations found in this study).

Furthermore, we demonstrate the importance of examining process, as opposed to product data whereby examining PLG informs us of the overall product, compared to investigating LSA scores, which reveal the process of how students' notes and summaries are accurate during learning with MetaTutor. By doing so, this allows us to assess how students' SRL unfolds over time. Additionally, the nature of our analysis included assessing the qualitative nature of students' notes and summaries as opposed to quantitatively examining frequency of cognitive strategies (e.g., [13]).

Lastly, our study also demonstrates the usefulness of including multichannel data to investigate cognitive, affective, and metacognitive SRL processes during learning with ITSs, as opposed to using traditional self-report measures that rely on students' subjective feelings that do not investigate behavioral actions (i.e., from log files and videos of facial expressions of emotions).

4.1 Future Directions: Towards Adaptive ITSs

Future research should develop ITSs that foster not only cognitive processes, such as taking notes and making summaries, but also affective processes, such as confusion during learning. Specifically, ITSs can assess the accuracy of notes and summaries by scoring their LSA in real time and providing these scores to students as adaptive scaffolding and feedback to improve the accuracy of strategy use. In addition, the ITS can also measure students' emotions in real time (e.g., [5]) and suggest the appropriate emotion regulation strategies (e.g., [14]) so students can express emotions that can enhance their note taking and summarizing (e.g., confusion). This can ensure that students are using the appropriate CAMM SRL processes during learning with ITSs.

Lastly, these results can further the development of ITSs that adapt based on students' cognitive and affective processes, such that the systems can foster SRL based on the accuracy of students' notes and summaries and how they might be impacted based on their levels of emotions. For example, based on findings from this study, we found a positive correlation between confusion and LSA score. Adaptive ITSs, therefore, can aim to foster confusion for students with low levels of it (i.e., by providing discrepancies in the text and diagram; [15]) to ensure their notes and summaries are accurate. Thus, this study can contribute to the overarching goal of developing ITSs: to foster effective SRL for all students.

Acknowledgements. This research was supported by funding from the National Science Foundation (DRL#1431552; DRL#1660878, DRL#1661202) and the Social Sciences and Humanities Research Council of Canada (SSHRC 895-2011-1006). The authors would like to thank the members from the SMART Lab at NCSU for their assistance with data collection.

References

1. Azevedo, R., Taub, M., Mudrick, N. V.: Understanding and reasoning about real-time cognitive, affective, and metacognitive processes to foster self-regulation with advanced learning technologies. In: Alexander, P.A., Schunk, D.H., and Greene, J.A. (eds.) Handbook of Self-regulation of Learning and Performance, 2nd ed., pp. 254–270. Routledge, New York (2018)
2. Biswas, G., Segedy, J.R., Bunchongchit, K.: From design to implementation to practice a learning by teaching system: Betty's brain. Int. J. Artif. Intell. Educ. **26**, 350–364 (2016)

3. Winne, P.H.: Cogniion and metacognition within self-regulated learning. In: Alexander, P.A., Schunk, D.H., Greene, J.A. (eds.) Handbook of Self-regulation of Learning and Performance, 2nd ed., pp. 36–48. Routledge, New York (2018)
4. D'Mello, S., Graesser, A.: Dynamics of affective states during complex learning. Learn. Instr. **22**, 145–157 (2012)
5. D'Mello, S., Graesser, A.C.: Feeling, thinking, and computing with affect-aware learning technologies. In: Calvo, R.A., D'Mello, S.K., Gratch, J., Kappas, A. (eds.) Handbook of Affective Computing, pp. 419–434. Oxford University Press, New York (2015)
6. Bonner, J.M., Holliday, W.G.: How college science students engage in note-taking strategies. J. Res. Sci. Teach. **43**, 786–818 (2006)
7. Witherspoon, A.M., Azevedo, R., D'Mello, S.: The dynamics of self-regulatory processes within self-and externally regulated learning episodes during complex science learning with hypermedia. In: Woolf, Beverley P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) ITS 2008. LNCS, vol. 5091, pp. 260–269. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-69132-7_30
8. Landauer, T., McNamara, D.S., Dennis, S., Kintsch, W.: Handbook of Latent Semantic Analysis. Erlbaum, Mahwah (2007)
9. Dente, P., Küster, D., Skora, L., Krumhuber, E.G.: Measures and metrics for automatic emotion classification via FACET. In: Bryson, J., De Vos, M., and Padget, J. (eds.) Proceedings of the Conference on the Study of Artificial Intelligence and Simulation of Behaviour (AISB), pp. 160–163 (2017)
10. Ekman, P., Friesen, W.V., Hager, J.C.: Facial Action Coding System. Network Information Research Corporation, Salt Lake City (2002)
11. Littlewort, G., Wu, T., Whitehill, J., Fasel, I., Movellan, J., Bartlett, M.: CERT computer expression recognition tool. In: Automatic Face and Gesture Recognition, pp. 298–305. IEEE, New York (2011)
12. D'Mello, S., Lehman, B., Pekrun, R., Graesser, A.: Confusion can be beneficial for learning. Learn. Instr. **29**, 153–170 (2014)
13. Taub, M., Azevedo, R., Bouchet, F., Khosravifar, B.: Can the use of cognitive and metacognitive self-regulated learning strategies be predicted by learners' levels of prior knowledge in hypermedia-learning environments? Comput. Hum. Behav. **39**, 356–367 (2014)
14. Azevedo, R., Taub, M., Mudrick, N.V., Millar, G.C., Bradbury, A.E., Price, M.J.: Using data visualizations to foster emotion regulation during self-regulated learning with advanced learning technologies. In: Buder, J., Hesse, F.W. (eds.) Informational Environments: Effects of Use, Effective Designs, pp. 225–247. Springer, Amsterdam, The Netherlands (2017). https://doi.org/10.1007/978-3-319-64274-1_10
15. Burkett, C., Azevedo, R.: The effect of multimedia discrepancies on metacognitive judgments. Comput. Hum. Behav. **28**, 1276–1285 (2012)

Adaptive Feedback Based on Student Emotion in a System for Programming Practice

Thomas James Tiam-Lee(✉) and Kaoru Sumi(✉)

Future University Hakodate, Hakodate, Hokkaido 041-8655, Japan
g3117002@fun.ac.jp, kaoru.sumi@acm.org

Abstract. We developed a system for programming practice that provides adaptive feedback based on the presence of confusion on the student. The system provides two types of adaptive feedback. First, it can control the complexity of the exercises presented to the student. Second, it can offer guides for the exercises when needed. These feedback are based on the presence of confusion, which is detected based on the student's compilations, typing activity, and facial expressions using a hidden Markov model trained on data collected from introductory programming course students. In this paper we discuss the system, the approach for detecting confusion, and the types of adaptive feedback displayed. We tested our system on Japanese university students and discuss the results and their feedback. This study can lay the foundation for the development of intelligent programming tutors that can generate personalized learning content based on the state of the individual learner.

Keywords: Affective feedback · Programming · Education

1 Introduction and Related Studies

Computer programming is now a core competence in various professions in the 21st century. According to a 2017 report by Digital Promise, several states in the USA have already taken steps to include computer science in K-12 education [13]. In a 2015 report by European Schoolnet, sixteen countries in the EU have already integrated coding in their curriculum in the national, regional, or local level [7]. Japan aims to include computer science education starting in primary schools by 2020 [4].

With the rise of the number of people learning how to code, there is also a need for the development of tools and resources that support programming education. Currently, there are several initiatives with varying goals that support learning programming. Online websites such as Codecademy [2] and Code.org [3] introduce coding to a wide audience. Applications such as Alice [12], Scratch [31], and Reduct [6] use visual representations of programming concepts to introduce children to coding. Intelligent programming tutors (IPT) such as Programming

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 243–255, 2018.
https://doi.org/10.1007/978-3-319-91464-0_24

Tutor [23], Code Adviser [5], ITAP [32], and Ask-Elle [19] provide customized feedback to students like a human tutor.

One characteristic of human tutors is their ability to respond to the emotions of learners. For example, a confused student may be given hints, while a student who is frustrated may be given empathy and encouragement. Emotions play an important role in the learning process. It is said that positive emotions can enhance memory performance and improve reasoning, while negative emotions can disturb the memory retrieval process [17]. Recently, some intelligent tutoring systems have been designed to respond not only to the student's cognitive state, but also their affective state, with promising results. In AutoTutor, a tutor avatar that responds to the emotion of the student during learning was shown to have better results in terms of knowledge transfer [15]. Similarly, in Wayang Tutor [38] and Genetics with Jean [35], interventions are given based on student affect, such as providing encouragement in the case of low confidence.

In the domain of intelligent programming tutors, work on providing feedback based on affective states is relatively limited. In the work of Grafsgaard et al., facial expressions and postures of students were analyzed while working on programming activities with emotions induced from interactions with a human tutor in another room [20–22]. Java Sensei is an intelligent programming tutor for Java that includes an affective module. The student's emotion is detected based on facial expressions using neural networks. Based on this a tutor avatar may give feedback, empathetic responses, or interventions [11]. This work only considered basic emotions such as joy and anger, and not academic emotions such as confusion and boredom.

Rodrigo et al. has shown that some emotions experienced by novice programming students during coding sessions are correlated with their midterm exam performance [33]. Confusion was found to have a significantly negative correlation with achievement, suggesting that students who experience sustained confusion throughout coding sessions are likely to perform poorly in exams [26]. Thus, it is important for tutors to recognize confusion and resolve it. In this paper, we discuss a system that can generate programming exercises and guides based on the presence of confusion on the student.

2 System with Adaptive Feedback Based on Emotion

We developed a system for coding practice that displays adaptive feedback based on the emotion of the student. Practice is crucial in learning programming. An international survey by Lahtinen et al. shows that computer science teachers and students perceive that programming is learned better in self-study sessions than in classroom lectures [24]. A study by Barros et al. shows that encouraging students to do more practice increased the retention rate of students in programming courses [8]. Moreover, learner-centered teaching setups like flipped classrooms are recently getting more and more attention, where students are expected to learn actively on their own while human teachers serve more as a guide than a source of information [18].

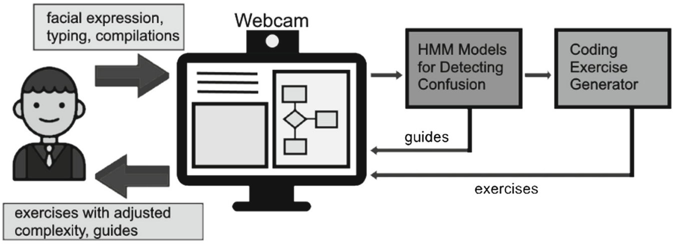
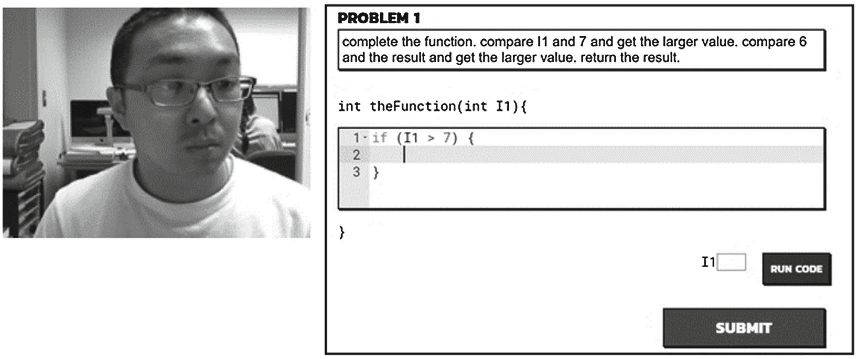

facial expression, typing, compilations
Webcam
HMM Models for Detecting Confusion
Coding Exercise Generator
guides
exercises
exercises with adjusted complexity, guides

Fig. 1. Diagram of system flow



PROBLEM 1

complete the function. compare I1 and 7 and get the larger value. compare 6 and the result and get the larger value. return the result.

```
int theFunction(int I1){
if (I1 > 7) {

}
}
```

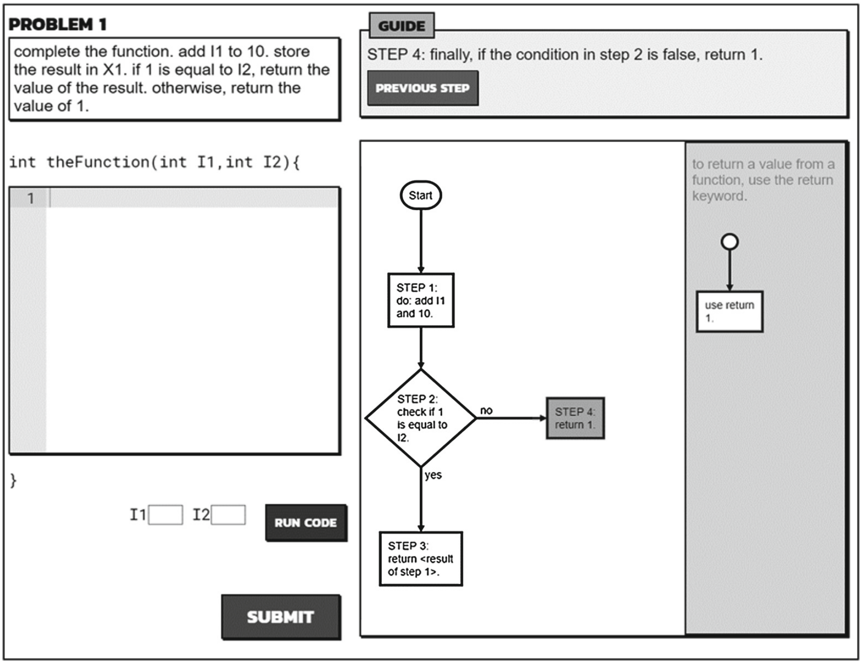
Fig. 2. Left: a student using the system, right: a screen shot of the system

Adaptive feedback is defined as a dynamic kind of feedback, wherein different learners receive different kinds of instruction or content based on certain factors [25]. In our system, adaptive feedback is based on the presence of confusion, which is detected based on the compilations, presence of typing, and facial expressions of the student. Currently, our system presents two kinds of adaptive feedback. First, the system can adjust the complexity of the exercises presented to the student. Second, the system can display exercise-specific guides in the form of a visualization.

Our system is intended to be used with a web camera, as it uses facial expression information as a feature in the detection of emotion. Figure 1 shows a diagram of the system flow and Fig. 2 shows a student using our system.

We used Java as the programming language because it is commonly-used in introductory programming classes. The system generates a series of coding exercises in which the student must write the body of a function that performs a specified task. The system provides an interface for the student to write code, test it by providing values for the function arguments, and submit it for checking. Once submitted, the system can automatically check the code by running it against a set of test cases and matching the return value of each test case with the expected return value. If the student submits a correct solution to the exercise, the system generates the next exercise. If the student is unable to solve an exercise for seven minutes, an option to give up and get an easier exercise becomes available.

According to cognitive disequilibrium theory [30], when a learner enters a state of confusion, a tutor should provide interventions to encourage the student to continue working and resolve the confusion before the student gives up [14]. In our system, the system detects the presence of confusion every 10 seconds using hidden Markov models trained from Japanese university students [36,37]. This detection process is discussed in more detail in Sect. 3. If confusion is detected, the system offers a guide to the student. If accepted, the system displays a visualization of the exercise, which shows the individual steps of the exercise presented in the form of a flowchart. The student can select an individual step and see a text hint of how to perform that step. Visualization has been used to improve student understanding in computer programming [9,28]. Figure 3 shows a screen shot of the system with a guide displayed for the exercise.



PROBLEM 1

complete the function. add I1 to 10. store the result in X1. if 1 is equal to I2, return the value of the result. otherwise, return the value of 1.

```
int theFunction(int I1,int I2){

}
```

GUIDE

STEP 4: finally, if the condition in step 2 is false, return 1.

to return a value from a function, use the return keyword.

PREVIOUS STEP
Start
STEP 1: do: add I1 and 10.
STEP 2: check if 1 is equal to I2.
no
STEP 4: return 1.
yes
STEP 3: return <result of step 1>.
use return 1.
I1
I2
RUN CODE
SUBMIT

Fig. 3. Screenshot of the system for programming practice with guide displayed

When the system starts, it generates a coding exercise. The complexity of the succeeding exercise is adjusted based on the student's acceptance of guide offers. We measure complexity as the number of operations needed to solve it. If the student solves two problems without accepting any guide offer, the complexity of the succeeding exercise is increased. If the student accepts a guide offer, the complexity remains the same. The complexity is decreased when the student gives up on an exercise. Adjusting content based on domain mastery is one of the factors in adaptive learning environments [29] and has been implemented in tutoring systems in other domains [27,34].

3 Detection of Student Emotion

In this section we discuss our approach for emotion detection. We used hidden Markov models built from compilations, typing activity and facial expressions to detect the presence of confusion.

3.1 Data Collection

We built models for recognizing confusion based on data collected from 11 Japanese freshmen students of Future University Hakodate. Each of the students has around 2 months of programming experience. All of them were taking a course on introductory programming during the time of the data collection.

Each test subject took part in the data collection process individually. Each participant was asked to solve a series of programming exercises of increasing difficulty. In each exercise, they had to write the body of a function that is supposed to perform a given task. The exercises covered introductory programming concepts which are variables, expressions, conditional statements, iterative statements, and arrays. Table 1 shows the exercises that were given during the session.

Table 1. Problems given to the students

No	Problem description
1	Display/return “Hello World”
2	Given the price and the money paid, compute the change
3	Given a temperature value in degrees Celsius, convert it to degrees Fahrenheit
4	Given a score, display/return “passed” if it is at least 60 or “failed” otherwise
5	Compute the amount due after applying rules for discounts
6	Compute the average of a list of scores
7	Given an integer, compute the sum of all the even digits
8	Determine if a number is prime or not

The session lasted for 45 min, or until the subject has solved all the exercises correctly. The subject was not allowed to move on to the next exercise until a correct solution has been submitted. The participants wrote their code in a special application. In this application, the student can write his code on the text editor, test his code by providing values for each function parameter, or submit the code for checking. The system checked if the submission is correct by running the code against a predefined set of test cases and checking if the return values match the expected outputs.

Throughout the entire session, all the keystrokes, compilations and a video feed of the student's face were logged. We used these data as features for detecting student emotion. Facial expressions have been used in several studies as an indicator of human emotion. Ekman, et al. presented a mapping of facial action units to basic emotions such as joy and anger [16]. Aside from facial expressions, we also added typing and compilation activity as possible indicators of emotion in a programming context, based on our idea that the occurrence of these events may vary depending on the student's emotion.

We were able to collect around 8 h of session data in total. From this, we reconstruct each session based on the logged keystrokes and video data so that the entire session could be replayed to the student. We asked each test subject to go through the entire session, mark time intervals and annotate them based on the emotion that they were feeling at that time. The student was given the freedom to decide which parts of the session they wanted to annotate. For this study, the labels we considered are based on the common emotions experienced by students during programming from the work of Bosch [10]. These are: engaged, confused, frustrated, and bored.

Majority of the reports were either "engaged" or "confused". This was probably because the short time of the data collection did not allow for confusion to prolong and transition to the other more undesirable emotions. Because of this, we focus on engagement and confusion only for this study. We consider engagement as the ideal state and confusion as the undesired state. Although we do not specifically detect other emotions such as frustration and boredom, we hypothesize that these emotions will more likely be classified as "confused" than "engaged".

3.2 Building Hidden Markov Models for Classifying Emotion

A total of 20 intervals labeled "confused" and 24 intervals labeled "engaged" were collected. We treated each interval as a Markov chain by dividing it into a sequence of discrete states. First, we treated code compilations as a compilation state. Next, parts of the interval where the student was typing were treated as a typing state. A threshold of 3 s was used to determine the boundaries of the interval (i.e., if the student did not type anything for 3 s that means that the "typing state" has already ended). Finally, all the remaining parts of the sequences were treated as idle states.

We treated the problem as a binary classification problem, "given a sequence, does it represent a time where the student is confused or engaged?" To build a classifier for this, we use all the sequences labeled as "confused" and trained a hidden Markov model (HMM) that represents confusion. We did the same thing for all sequences labeled as "engaged", training another HMM that represents engagement. The best fit model for each classification was chosen based on the number of hidden states that yielded the highest likelihood. Using the two HMM models, an unknown sequence can be classified by computing for the likelihood that it was generated by the confusion model and the engagement model and choosing which yields the higher value. We repeated this process again with the facial expression information. Figure 4 shows an example HMM sequence.

To get the facial expressions from the video feeds, we use Affectiva SDK [1] to automatically extract the Facial Action Coding System (FACS) points from the video and recognize the presence of different action units (AU). The FACS points are critical points that indicate facial muscle movement. Certain configurations of FACS points correspond to an action unit, which represents a single unit of movement such as raising the eyebrow or opening the mouth. Affectiva processes a video feed frame by frame and uses its model to infer the presence of different AUs by assigning integer scores from 0 to 100, representing its confidence that each AU has occurred in that frame.

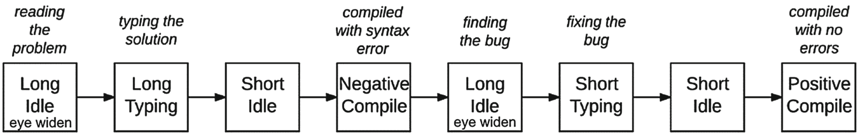


Fig. 4. An example state sequence from the collected data. The text in italic describe a possible scenario that could be happening at that time

We found that the most common AUs that occurred in general across all subjects are dimpler (AU16), lip press (AU24), lip suck (AU28), eye widen (AU5), and mouth open (AU27). Dimpler refers to the tightening of the corners of the lips. Lip press refers to pressing the lips together. Lip suck refers to pulling the lips and sucking the adjacent skin in the mouth. Eye widen refers to raising the upper eyelids such that the eyes appear bigger than normal. Mouth open refers to lowering the lower lip such that the lips are not touching one another. Examples of these facial expressions are shown in Fig. 5. These AUs in discussed more detail in [36,37].

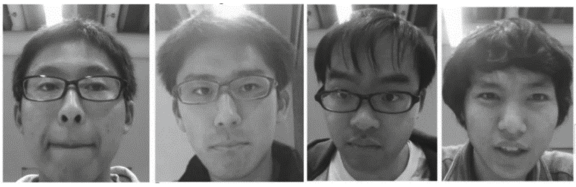


Fig. 5. From left to right: lip suck, lip press, eye widen, and mouth open

We incorporated these facial expressions into the classification process by training five pairs of HMM models, one pair for each AU. We follow the same process discussed above, but for each idle and typing state, we differentiated between those states where the AU has been observed at least once, and where the AU has not been observed (i.e., instead of having one state for long idle, we now have a state for long idle with the AU, and one state for long idle without the AU). We say that the AU has been observed if Affectiva's assigns a score of more than 50 to that AU in any of the frames within that interval. This increases the number of discrete states from 6 to 10.

Table 2 shows the accuracy of the models using leave one out cross fold validation. True positive (TP) refers to the number of intervals labeled as confused that are correctly classified as confused. False positive (FP) refers to the number of intervals labeled as engaged but incorrectly classified as confused. True negative (TN) refers to the number of intervals labeled as engaged that are correctly classified as engaged. False negative (FN) refers to the number of intervals labeled as confused but are incorrectly classified as engaged.

Table 2. Results of leave-one-out cross fold validation

AU	TP	FP	TN	FN	Accuracy	Kappa
no AU information	14	10	14	6	63.64%	0.28
dimpler (AU16)	13	7	17	7	68.18%	0.35
lip press (AU24)	12	6	18	8	68.18%	0.35
lip suck (AU28)	13	9	15	7	63.64%	0.27
eye widen (AU5)	9	10	11	14	45.46%	−0.08
mouth open (AU27)	15	8	16	5	70.46%	0.41

In the system for programming practice, to detect confusion in the system for unknown state sequences, we represent the unknown sequence of states and feed it to each of the five pairs of HMM models (dimpler, lip press, lip suck, eye widen, and mouth open) and get the majority result.

4 Evaluation

In this section we discuss the evaluation of the system. The purpose of the evaluation is to determine whether the exercises that were generated can be helpful in programming practice, and whether adaptive feedback based on the presence of confusion has a positive impact on the learning experience.

4.1 Experiment Design

We tested our system on 35 Japanese university students from Future University Hakodate, Japan. The students were from different year levels, from freshmen students to graduate school students. The students were divided into two groups, labeled Mode A (17 students) and Mode B (18 students). Students were divided such that each group had a balanced representation in terms of year level, age, sex, and months of programming experience. Each student was asked to use the system for 40 min. Students in Mode A used a version of the system that does not offer any guide, even when confusion is detected, and the problems generated were of random complexity. Students in Mode B used a version of the system that offers a guide whenever confusion is detected. Furthermore, the complexity of the exercises was controlled based on the guide acceptance as described in Sect. 2. The system was translated to Japanese for the evaluation.

4.2 Results and Discussion

At the end of the session, we asked the students to rate the exercises on a Likert scale from 1 to 5, with 1 being "very much", and 5 being "not at all" based on how fun the exercises were, and how helpful the exercises were in practicing programming. For each criterion, students can also leave optional qualitative feedback. Table 3 shows the responses of the subjects.

Table 3. Responses on how fun and how helpful the exercises were

Group	How fun?					How helpful?				
	1	2	3	4	5	1	2	3	4	5
Mode A	2	7	3	5	0	2	8	4	3	0
Mode B	4	9	2	3	0	2	12	3	1	0
Total	6	16	5	8	0	4	20	7	4	0

62.68% of the test subjects across the two groups thought that the exercises were "very fun" or "fun". There were more students in Mode B who thought the exercises were fun, but the difference among the two groups is not significant. It should be noted that some students in Mode B cited the progression of the exercises from easy to difficult as a factor for how fun it was to answer them, thus showing the importance of progression in student engagement. In general, students who thought the exercises were not fun cited the fact that the problems were not interesting enough or felt too similar with one another. 68.57% of the students across the two groups thought that the exercises are "very helpful" or "helpful" in programming practice. The most common feedback included how the exercises helped them review the concepts that they already forgot.

Students in Mode B solved considerably more problems than those in Mode A. A 90% trimmed mean shows that students in Mode A solved an average of 6.33 problems, while students in Mode B solved an average of 13.5 problems. Furthermore, students in Mode A clicked the "give up" button more times than students in Mode B. In total, students gave up 36 times in Mode A, but only 10 times in Mode B. These show that the adaptive feedback used in Mode B has had some effect on the number of exercises that were successfully solved by the students, although it is unclear how much of this is contributed by the guides and how much is contributed by the controlled problem complexity.

To evaluate the detection of confusion, we logged the instances in the session where the system detected confusion. If confusion was detected multiple times in a single exercise, we only considered the first instance. We showed these a replay of each of those instances starting from 30 s before the confusion was detected up to the point that it was detected. We then ask the student what he felt at that time, to which they could respond "very confused", "somewhat confused", or "not confused at all". To avoid biased responses, the student was not informed that these points in the session were points where the system detected confusion. Overall, 77.78% of all the instances where the system detected confusion match the actual emotion of the student ("very confused" or "somewhat confused"), as shown in Table 4 (left).

Table 4 (right) shows the number of times the students in Mode B accepted the offer for a guide. In moments where students were very confused, they were very likely to accept the guide that was offered (82.61%). Interestingly, in 65% of the instances where students reported that they were not confused at all, they also accepted the offer of a guide. Most of these instances happened in the earlier parts of the session, so it was possible that the students were simply curious to see what the guides are like.

Table 4. Students actual emotions in times where system detected confusion (left) and acceptance rate of guides (right). VC, SC, and NC stand for "very confused", "somewhat confused", and "not confused" respectively. A and NA stand for "accepted" and "not accepted" respectively.

	VC	**SC**	**NC**
Mode A	26	19	10
Mode B	23	37	20
Total	49	56	30

	A	**NA**	**Total**
VC	19	4	23
SC	19	18	37
NC	13	7	20
Total	51	29	

Out of the 38 times the students accepted a hint when they felt "very confused" or "somewhat confused", a correct submission was made within 3 min since the guide was accepted 26 (68.42%) times. On the other hand, the student eventually gave up in 6 instances (15.79%) even after the guide was accepted. This shows that the hints, in most instances, were effective in resolving student confusion. 16 out of the 17 students (94.12%) who received hints said that while they did not learn anything new, the system was helpful in enabling them to practice on what they already know. 1 student (5.89%) said the hints were not helpful but did not provide any reason why.

Overall, our evaluation shows that adaptive feedback based on emotion has a potential in helping students learn or practice coding. Majority of the students reported having fun in solving the exercises and reported that the exercises can be helpful in programming practice, and the guides offered show positive effects on the experience of each student.

5 Conclusion

In this paper we have presented a system for coding practice that provides adaptive feedback based on the presence of confusion on the student. We show that there is potential for such kind of feedback to positively affect the experience of students in practicing programming.

There are several directions that could be explored for future work on this study. We made several assumptions in this study. Complexity was defined to be the number of operations in the exercise, which may not be the best indicator of complexity in practice. Confusion was also addressed in a general sense. In succeeding studies, different types of confusion can be identified to present more effective feedback. Other affective states such as frustration and boredom could also be considered, as well as other types of feedback.

References

1. Affectiva developer portal. https://developer.affectiva.com/. Accessed 04 Jan 2018
2. Codecademy. https://www.codecademy.com. Accessed 04 Jan 2018
3. Code.org. https://code.org. Accessed 04 Jan 2018
4. Programming education at elementary school level - ministry of education, culture, sports, science and technology Japan. http://www.mext.go.jp/b_menu/shingi/chousa/shotou/122/attach/1372525.htm. Accessed 04 Jan 2018
5. Ade-Ibijola, A., Ewert, S., Sanders, I.: Introducing code adviser: a DFA-driven electronic programming tutor. In: Drewes, F. (ed.) CIAA 2015. LNCS, vol. 9223, pp. 307–312. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-22360-5_25
6. Arawjo, I., Wang, C.Y., Myers, A.C., Andersen, E., Guimbretière, F.: Teaching programming with gamified semantics. In: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, pp. 4911–4923. ACM (2017)
7. Balanskat, A., Engelhardt, K.: Computer programming and coding: priorities, school curricula and initiatives across Europe, European schoolnet (2015)
8. Barros, J.P., Estevens, L., Dias, R., Pais, R., Soeiro, E.: Using lab exams to ensure programming practice in an introductory programming course. ACM SIGCSE Bull. **35**(3), 16–20 (2003)
9. Ben-Ari, M.: Visualization of programming. Improv. Comput. Sci. Educ. **52** (2013)
10. Bosch, N., D'Mello, S., Mills, C.: What emotions do novices experience during their first computer programming learning session? In: Lane, H.C., Yacef, K., Mostow, J., Pavlik, P. (eds.) AIED 2013. LNCS (LNAI), vol. 7926, pp. 11–20. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39112-5_2
11. Cabada, R.Z., Estrada, M.L.B., Hernández, F.G., Bustillos, R.O.: An affective learning environment for Java. In: 2015 IEEE 15th International Conference on Advanced Learning Technologies (ICALT), pp. 350–354. IEEE (2015)
12. Cooper, S., Dann, W., Pausch, R.: Alice: a 3-D tool for introductory programming concepts. J. Comput. Sci. Coll. **15**, 107–116 (2000). Consortium for Computing Sciences in Colleges

13. Digital Promise: Computational thinking for a computational world (2017)
14. DMello, S., Jackson, T., Craig, S., Morgan, B., Chipman, P., White, H., Person, N., Kort, B., el Kaliouby, R., Picard, R., et al.: Autotutor detects and responds to learners affective and cognitive states. In: Workshop on Emotional and Cognitive Issues at the International Conference on Intelligent Tutoring Systems, pp. 306–308 (2008)
15. DMello, S.K., Lehman, B., Graesser, A.: A motivationally supportive affect-sensitive autotutor. In: Calvo, R., D'Mello, S. (eds.) New Perspectives on Affect and Learning Technologies, vol. 3, pp. 113–126. Springer, Heidelberg (2011). https://doi.org/10.1007/978-1-4419-9625-1_9
16. Ekman, P., Friesen, W.V.: Unmasking the face: a guide to recognizing emotions from facial cues (1975)
17. Frasson, C., Chalfoun, P.: Managing learners affective states in intelligent tutoring systems. In: Nkambou, R., Bourdeau, J., Mizoguchi, R. (eds.) Advances in Intelligent Tutoring Systems. SCI, vol. 308, pp. 339–358. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-14363-2_17
18. Fulton, K.: Upside down and inside out: flip your classroom to improve student learning. Learn. Leading Technol. **39**(8), 12–17 (2012)
19. Gerdes, A., Heeren, B., Jeuring, J., van Binsbergen, L.T.: Ask-elle: an adaptable programming tutor for haskell giving automated feedback. Int. J. Artif. Intell. Educ. **27**(1), 65–100 (2017)
20. Grafsgaard, J.F., Boyer, K.E., Lester, J.C.: Predicting facial indicators of confusion with hidden Markov models. In: D'Mello, S., Graesser, A., Schuller, B., Martin, J.-C. (eds.) ACII 2011. LNCS, vol. 6974, pp. 97–106. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-24600-5_13
21. Grafsgaard, J.F., Wiggins, J.B., Boyer, K.E., Wiebe, E.N., Lester, J.C.: Automatically recognizing facial indicators of frustration: a learning-centric analysis. In: 2013 Humaine Association Conference on Affective Computing and Intelligent Interaction (ACII), pp. 159–165. IEEE (2013)
22. Grafsgaard, J.F., Wiggins, J.B., Boyer, K.E., Wiebe, E.N., Lester, J.C.: Embodied affect in tutorial dialogue: student gesture and posture. In: Lane, H.C., Yacef, K., Mostow, J., Pavlik, P. (eds.) AIED 2013. LNCS (LNAI), vol. 7926, pp. 1–10. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39112-5_1
23. Keuning, H., Heeren, B., Jeuring, J.: Strategy-based feedback in a programming tutor. In: Proceedings of the Computer Science Education Research Conference, pp. 43–54. ACM (2014)
24. Lahtinen, E., Ala-Mutka, K., Järvinen, H.M.: A study of the difficulties of novice programmers. In: ACM Sigcse Bulletin, vol. 37, pp. 14–18. ACM (2005)
25. Le, N.T.: A classification of adaptive feedback in educational systems for programming. Systems **4**(2), 22 (2016)
26. Lee, D.M.C., Rodrigo, M.M.T., Baker, R.S.J., Sugay, J.O., Coronel, A.: Exploring the relationship between novice programmer confusion and achievement. In: D'Mello, S., Graesser, A., Schuller, B., Martin, J.-C. (eds.) ACII 2011. LNCS, vol. 6974, pp. 175–184. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-24600-5_21
27. Melis, E., Andres, E.: Global feedback in activemath. J. Comput. Math. Sci. Teach. **24**(2), 197 (2005)
28. Myers, B.A.: Taxonomies of visual programming and program visualization. J. Vis. Lang. Comput. **1**(1), 97–123 (1990)

29. Okpo, J., Masthoff, J., Dennis, M., Beacham, N.: Conceptualizing a framework for adaptive exercise selection with personality as a major learner characteristic. In: Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization, pp. 293–298. ACM (2017)
30. Piaget, J., Cook, M.: The Origins of Intelligence in Children, vol. 8. International Universities Press, New York (1952)
31. Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B., et al.: Scratch: programming for all. Commun. ACM **52**(11), 60–67 (2009)
32. Rivers, K., Koedinger, K.R.: Data-driven hint generation in vast solution spaces: a self-improving python programming tutor. Int. J. Artif. Intell. Educ. **27**(1), 37–64 (2017)
33. Rodrigo, M.M.T., Baker, R.S., Jadud, M.C., Amarra, A.C.M., Dy, T., Espejo-Lahoz, M.B.V., Lim, S.A.L., Pascua, S.A., Sugay, J.O., Tabanao, E.S.: Affective and behavioral predictors of novice programmer achievement. In: ACM SIGCSE Bulletin, vol. 41, pp. 156–160. ACM (2009)
34. Salden, R.J., Paas, F., Van Merriënboer, J.J.: Personalised adaptive task selection in air traffic control: effects on training efficiency and transfer. Learn. Instr. **16**(4), 350–362 (2006)
35. Thompson, N., McGill, T.J.: Genetics with jean: the design, development and evaluation of an affective tutoring system. Educ. Technol. Res. Dev. **65**(2), 279–299 (2017)
36. Tiam-Lee, T.J., Sumi, K.: Analyzing facial expressions and hand gestures in filipino students' programming sessions. In: 2017 International Conference on Culture and Computing (Culture and Computing), pp. 75–81. IEEE (2017)
37. Tiam-Lee, T.J., Sumi, K.: A comparison of Filipino and Japanese facial expressions and hand gestures in relation to affective states in programming sessions. In: Workshop on Computation: Theory and Practice 2017 (2017)
38. 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)

Learning by Explaining to a Digital Doppelganger

Ning Wang[1](✉), Ari Shapiro[1], Andrew Feng[1], Cindy Zhuang[2], Chirag Merchant[1], David Schwartz[2], and Stephen L. Goldberg[3]

[1] Institute for Creative Technologies, University of Southern California, Los Angeles, USA
{nwang,shapiro,feng,merchant}@ict.usc.edu
[2] Department of Psychology, University of Southern California, Los Angeles, USA
cindyi.0324@gmail.com, davschw@dornsife.usc.edu
[3] U.S. Army Research Laboratory, Orlando, USA
stephen.l.goldberg.civ@mail.mil

Abstract. Digital doppelgangers are virtual humans that highly resemble the real self but behave independently. An emerging computer animation technology makes the creation of digital doppelgangers an accessible reality. This allows researchers in pedagogical agents to explore previously unexplorable research questions, such as how does increasing the similarity in appearance between the agent and the student impact learning. This paper discusses the design and evaluation of a digital doppelganger as a virtual listener in a learning-by-explaining paradigm. Results offer insight into the promise and limitation of this novel technology.

Keywords: Pedagogical agent · Learning by explaining
Rapid Avatar Capture and Simulation

1 Introduction

Pedagogical agents are embodied animated virtual characters designed to help students learn [1]. Over the past two decades, since Herman the Bug [2] and Steve [3], researchers have studied many aspects of pedagogical agents, including animation [4], gesture [5], voice [6], and social intelligence [7], and role [8], to facilitate student learning across a great number of domains. As embodied virtual characters, one of the first decisions pedagogical agent designers have to make is what the agent looks like. Research on a pedagogical agent's appearance has indicated the impact of such design decisions on learning outcome [9], including recall [10] and transfer of learning [11] (for review see [12]). Research into appearance similarity between the agent and the learner mainly focused on ethnicity and behaviors consistent with such appearance (e.g., the use of dialect) [13]. Research questions further along the dimension of agent similarity with the learner have been left largely unanswered because of the need to generate

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 256–264, 2018.
https://doi.org/10.1007/978-3-319-91464-0_25

such agents for a large enough population and at sufficient speed to accommodate experiment sessions of limited duration. An emerging technology, the Rapid Avatar Capture and Simulation (RACAS) system, enables low-cost and high-speed scanning of a human user and creation of a fully animatable virtual 3D “digital double” of the user. This allows researchers to explore a previously unexplored research question: how does increasing the similarity in appearance between the agent and student impact learning. In this paper, we discuss the design of a digital doppelganger as a virtual listener and the evaluation of such an agent in a learning-by-explaining paradigm.

2 Explaining to a Digital Doppelganger

Digital doppelgangers are virtual humans that highly resemble the real self but behave independently [14]. The RACAS system, described in detail in [15], makes the digital doppelganger a more accessible reality. We designed a virtual listener and incorporated digital doppelgangers created by RACAS to embody the listener. A human speaker can converse with the agent and the agent can respond with conversational backchannel feedback [16]. The feedback is generated based on analysis of the speaker’s nonverbal behavior, such as head nods, prosody, etc. [16]. Previous research has shown the value of such feedback in creating rapport with the human speaker [16]. The current work focuses specifically on examining the impact of agent appearance on measures related to student learning. We hypothesize that teaching a virtual listener who looks just like oneself can impact a learner’s motivation and self-regulation in learning (e.g. persisting in a learning task), and ultimately improve learning outcomes. Specifically, we hypothesize that, in a learning-by-explaining paradigm:

H1: A virtual listener that shares the appearance of the learner can improve learner motivation to teach the agent.

H2: A virtual listener that shares the appearance of the learner can improve student learning of domain knowledge through teaching the virtual agent.

H3: A virtual listener that shares the appearance of the learner can improve student self-efficacy through teaching the virtual agent.

3 Evaluation

Design. We conducted a study with the digital doppelganger serving as a virtual listener in the task of learning-by-explaining. In this task, a student first reads a passage on the human circulatory system, then verbally explains the topic to the virtual listener. The study is a between-subject design with two experiment conditions: the Digital Doppelganger condition and the virtual human condition.

- **Digital Doppelganger** In this condition, a virtual listener was constructed at the beginning of each experiment session using RACAS, thus sharing the appearance of the participants.

- **Virtual Human** In this condition, a virtual listener with a photo-realistic appearance not based on the participant was used. To control the realism of the virtual listener used in both conditions, the agent in this condition was generated using captures of non-participants obtained with RACAS through the same process used in the other condition (Fig. 1). The virtual listener was gender-matched to the participant, e.g., male participants interacted with a male virtual human. Aside from the difference in appearance, both virtual listeners responded to the participants with the same behaviors, described in Sect. 2.

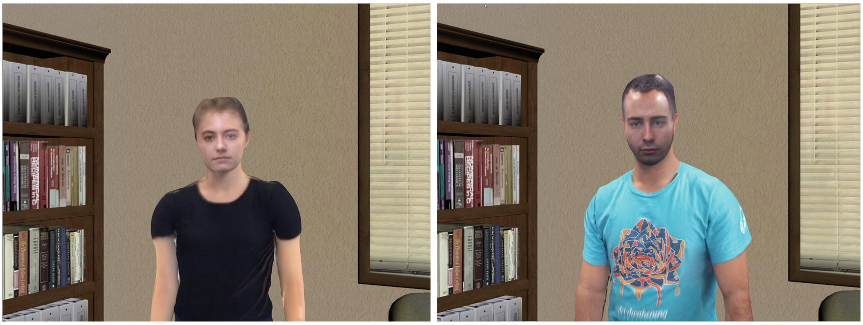


Fig. 1. Virtual human listeners, captured using RACAS, from the control condition.

Population and Procedure. We recruited 41 student either from the Psychology Department subject pool (received course credit) or via fliers posted on campus (received $10) at the University of Southern California. Participants first read an informed consent. Then the experimenter completed face and body scans of the participants, in both conditions. The full-body scan was captured with an iPad equipped with a specialized structure sensor. The face scan was captured using an Intel webcam with depth sensors. Next, the participants filled out a Background Survey and Pre-Test, then read a tutorial on the human circulatory system (adopted from [17]) on a web browser. The participants were told that they would later have to teach the material to a virtual student. Then, the participants sat in front of a 30-inch computer monitor with the display of the virtual student, and were told that the virtual student would represent him/her in a competition against other virtual students in a quiz on the same subject. Two cameras were fitted on top of the monitor: one recorded the participants' face, and the other served as input to the virtual listener. Participants then verbally explained what they had learned from the tutorial to the virtual listener. Finally, the participants filled out a Post-Interaction Survey and Post-Test. Each session was designed to last one hour.

Measures. The Background Survey included measures of demographic, education, Rosenberg Self-Esteem Scale [18], Adolescent Body Image Satisfaction Scale (ABISS) [19], Anxiety scale [20], and Self-Efficacy in domain knowledge (designed by the research team). The Self-Efficacy scale included items such as "If there is a quiz on human circulatory system, I expect to do well on the quiz". The Post-Interaction Survey included measures of Presence (constructed using items from [21,22]), Avatar Similarity ("To what extend do you feel that the virtual avatar resembled you?"), Desired Avatar Similarity ("If you had to design your own avatar for this task, how similar to your real appearance would you make your avatar?"), a repeated measure of Self-Efficacy in domain knowledge, and Self-Efficacy in the virtual student ("I think the avatar I just taught will do well in the competition."). In the Pre-Test, participants were asked to described 10 concepts on the human circulatory system and the path of blood through the body. The Post-Test included the Pre-Test questions and questions adopted from previous studies on human tutoring [17].

4 Results

Data from all 41 participants (26 female, 15 male, $M_{age} = 21.5$, age range: 19.7–29.7 years) are included in the analysis. The participants came from a variety of majors, ranging from psychology to fine arts, to biology, and many more. One participant had a graduate degree, while all other participants had some college education. Participants were randomly assigned to an experiment condition. While a balanced assignment was desired, in the end, 17 participants were assigned to the Digital Doppelganger condition and 24 to the Virtual Human.

Learning Domain Knowledge. An expert on the human circulatory system from the research team graded the Pre- and Post-Tests. On the Post-Test, we separated the score on questions that were repeated from the Pre-Test (Post-Test-Repeat) and scores on the rest of the questions (Post-Test-NonRepeat). We conducted an ANOVA with scores on Pre-Test and Post-Test-Repeat as a repeated measure and the experiment conditions as the Between-Subject factor. The result shows that there was a significant within-subject effect between Pre- and Post-Tests ($p < .001, F = 91.404$), while the between-subject effect due to the experiment manipulation was not statistically significant ($p = .308, F = 1.069$, see Fig. 2 for means). Although there is a noticeable difference on Pre-Test scores between the two experiment conditions, the difference is not statistically significant ($p = .308$). We also conducted an Independent Sample T-Test on the scores on Post-Test-NonRepeat and found no significant difference ($p = .821, M_{VH} = 32.33, M_{DD} = 33.12$, 62 total points available). This suggests that Hypothesis 2 regarding agent appearance and learning of domain knowledge is not supported.

Self-Efficacy. We conducted an ANOVA with self-efficacy before and after the study as the repeated measure and experiment condition as the Between-Subject

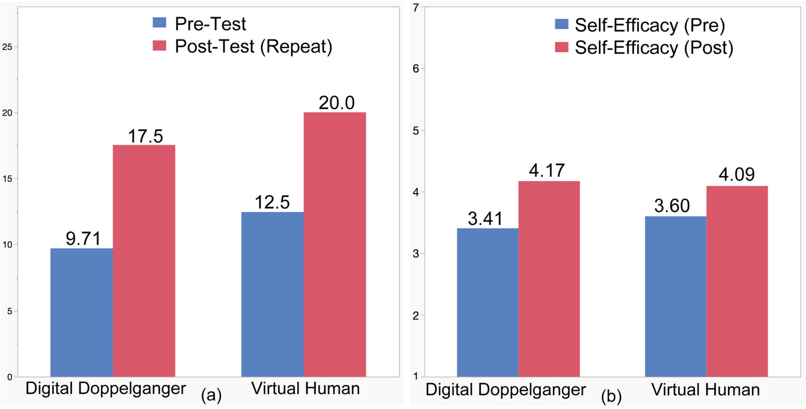


Fig. 2. (a) Comparison of Pre-Test and Post-Test-Repeat scores (maximum score was 28) (b) Comparison of Self-Efficacy (7-point Likert scale) before and after study between experiment conditions.

factor. The result shows that there was a significant within-subject effect before and after the study ($p = .003, F = 9.78$), while the between-subject effect due to the experiment manipulation was not statistically significant ($p = .891, F = .019$, see Fig. 2 for means). Additionally, we analyzed the participants' self-efficacy in the virtual listener, whom they taught and thought would represent them to compete with other agents. Again, we did not find any significant difference between the two experiment conditions ($p = .561, M_{VH} = 3.17, M_{DD} = 3.53$). This result suggests that Hypothesis 3, regarding the similarity of agent appearance and learner's self-efficacy, is not supported.

Motivation to Teach the Virtual Listener. We analyzed the time participants spent explaining the material to the virtual listener. An Independent Sample T-Test shows that there is no significant difference between the two conditions ($p = .105, M_{VH} = 277.88, M_{DD} = 208.63, Min = 55, Max = 645$, in seconds). This result suggests that Hypothesis 1 regarding the similarity of agent appearance and motivation to learn (and to explain and teach) is not supported.

Further Analysis. Because the results suggest that there is no statistically significant difference between the two experiment conditions, we conducted further analyses to examine why that was the case. We first performed a "manipulation check" on the Avatar Similarity scale. We expected the Avatar Similarity to be much lower in the Virtual Human condition, compared to the Digital Doppelganger condition. Independent-sample T-Test shows that it is indeed the case ($p = .004$). Figure 3 shows that participants from the Virtual Human condition did not perceive the agent's appearance to be similar to themselves. However, participants from the Digital Doppelganger condition did not think the virtual listener looked like them either (rated 3.76 on a 7-point Likert scale). Furthermore, we compared the Desired Avatar Similarity. The difference, as shown in Fig. 3, is marginally significant ($p = .077$). In particular, participants in the Virtual Human condition, who did not see their digital doppelganger, wished the virtual listener would look like them. Conversely, participants in the Digital Doppelganger condition, after seeing their own image manifested as an animated character, reported that they would rather the virtual listener *not* look like them.

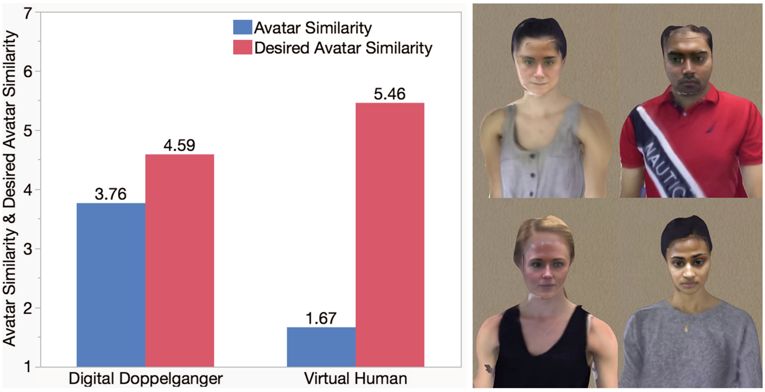


Fig. 3. Left: Comparison of the perceived similarity of the virtual listener's appearance to the participant's and the participant's desired level of such resemblance (7-point Likert scale). **Right:** Digital Doppelgangers that had pronounced imperfections, such as lighting, face mis-alignment, and missing pixels.

We then conducted pair-wise correlation tests of these two variables and the dependent variables we tested for the main hypothesis. The Desired Avatar Similarity is positively correlated with post-interaction Self-Efficacy ($r = .334, p = .033$), but not with the other dependent variables. This indicates that participants who were more confident in their domain knowledge had a higher desire for the virtual student to share their appearance. This resonates with the results on general self-confidence and confidence in one's appearance: the Desired Avatar Similarity is positively correlated with the Rosenburg Self-esteem measure ($r = .399, p = .01$) and the self-image measure—ABISS ($r = .436, p = .004$). The perceived Avatar Similarity, on the other hand, is positively correlated with the post-interaction Self-Efficacy in the agent ($r = .359, p = .021$), but not with the other dependent variables. This indicates that the more the participants perceived the agent to resemble themselves, the more confident they felt about how well the agent, whom they taught, would do in competitions and quizzes.

5 Discussion

In this paper, we discussed the design of a pedagogical agent for the learning-by-explaining paradigm. We applied a novel character-animation technology, RACAS, to create agents that share the physical appearance of a human learner. Evaluation of such agents showed that such resemblance did not significantly impact student learning of domain knowledge, their motivation to teach the agent, or their own self-efficacy. Further analysis indicates that when students are confident about their knowledge, they would like the agent to look like them. And the more the agent shared their appearance, the more confident they felt about the agent's future performance, as a result of their teaching. While the investigation did not yield a statistically significant result, it is worth noting that this is the first investigation of its kind. The process to scan, reconstruct, and animate a virtual agent, particularly one with an animatable face, in such rapid fashion has rarely been attempted before. The pedagogical agents created through such process are understandably less than perfect (see Fig. 3). Even very slight glitches in the virtual agent's appearance (e.g., misalignment of face and body) or animation (e.g., slight shift of the face when the eyes open/close) can distract the learner and interfere with engagement in the learning task.

The interaction with the digital doppelganger is short. Thus a novelty effect may have played a role in the study. Participants, especially the ones in the Digital Doppelganger condition who had never seen themselves transformed into a digital character before, may have directed much of their attention to visually inspecting their own avatar. Such activity, again, may have distracted the participants from the learning activity, both the recalling and the explaining. The distractions may have ultimately impacted the learning outcome. Future studies can allow learners to interact with their own avatar for longer periods of time, beyond the initial influence of the novelty effect. Additionally, previous studies on virtual listener agents have identified behavioral indications of when participants were distracted by the agent's behavior, e.g., speech disfluencies and gaze aversions. Linguistic and video analyses can be carried out on the participants' explanations and videos of their face to test this hypothesis on distraction. Since the study concluded, great improvements have already been made to RACAS that allow even faster capture of higher fidelity and more accurate 3D scans [23], all of which provide great promise for future studies on the appearance of pedagogical agents.

Acknowledgment. This work is supported by the US Army Research Laboratory. Statements expressed do not necessarily reflect the position or the policy of the United States Government. No official endorsement should be inferred.

References

1. Johnson, W.L., Rickel, J.W., Lester, J.C., et al.: Animated pedagogical agents: face-to-face interaction in interactive learning environments. Int. J. Artif. Intell. Educ. **11**(1), 47–78 (2000)
2. 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: Proceedings of Human Factors in Computing Systems, pp. 359–366. ACM (1997)
3. Johnson, W.L., Rickel, J.: Steve: an animated pedagogical agent for procedural training in virtual environments. SIGART **8**(1–4), 16–21 (1997)

4. Lester, J.C., Stone, B.A.: Increasing believability in animated pedagogical agents. In: Proceedings of Autonomous Agents, pp. 16–21. ACM (1997)
5. Craig, S.D., Gholson, B., Driscoll, D.M.: Animated pedagogical agents in multimedia educational environments: effects of agent properties, picture features and redundancy. J. Educ. Psychol. **94**(2), 428 (2002)
6. Person, N.K.: Autotutor improves deep learning of computer literacy: is it the dialog or the talking head? AI Educ. **97**, 47 (2003)
7. Wang, N., Johnson, W.L., Mayer, R.E., Rizzo, P., Shaw, E., Collins, H.: The politeness effect: pedagogical agents and learning outcomes. Int. J. Hum.-Comput. Stud. **66**(2), 98–112 (2008)
8. Biswas, G., Jeong, H., Kinnebrew, J.S., Sulcer, B., Roscoe, R.: Measuring self-regulated learning skills through social interactions in a teachable agent environment. Res. Practice Tech. Enhanced Learn. **5**(02), 123–152 (2010)
9. Baylor, A.L., Kim, Y.: Pedagogical agent design: the impact of agent realism, gender, ethnicity, and instructional role. In: Lester, J.C., Vicari, R.M., Paraguaçu, F. (eds.) ITS 2004. LNCS, vol. 3220, pp. 592–603. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-30139-4_56
10. Veletsianos, G.: Contextually relevant pedagogical agents: visual appearance, stereotypes, and first impressions and their impact on learning. Comput. Educ. **55**(2), 576–585 (2010)
11. Domagk, S.: Do pedagogical agents facilitate learner motivation and learning outcomes? J. Media Psychol. **22**, 84–97 (2010)
12. Schroeder, N.L., Adesope, O.O., Gilbert, R.B.: How effective are pedagogical agents for learning? A meta-analytic review. J. Educ. Comput. Res. **49**(1), 1–39 (2013)
13. Finkelstein, S., Yarzebinski, E., Vaughn, C., Ogan, A., Cassell, J.: The effects of culturally congruent educational technologies on student achievement. In: Lane, H.C., Yacef, K., Mostow, J., Pavlik, P. (eds.) AIED 2013. LNCS (LNAI), vol. 7926, pp. 493–502. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39112-5_50
14. Bailenson, J.N.: Doppelgangers-a new form of self? Psychologist **25**(1), 36–38 (2012)
15. Shapiro, A., Feng, A., Wang, R., Li, H., Bolas, M., Medioni, G., Suma, E.: Rapid avatar capture and simulation using commodity depth sensors. Comput. Anim. Virtual Worlds **25**(3–4), 201–211 (2014)
16. Gratch, J., Wang, N., Okhmatovskaia, A., Lamothe, F., Morales, M., van der Werf, R.J., Morency, L.-P.: Can virtual humans be more engaging than real ones? In: Jacko, J.A. (ed.) HCI 2007. LNCS, vol. 4552, pp. 286–297. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-73110-8_30
17. Chi, M.T., Siler, S.A., Jeong, H., Yamauchi, T., Hausmann, R.G.: Learning from human tutoring. Cogn. Sci. **25**(4), 471–533 (2001)
18. Rosenberg, M.: Society & the Adolescent Self-image. Princeton University Press, Princeton (2015)
19. Leone, J.E., Mullin, E.M., Maurer-Starks, S.S., Rovito, M.J.: The adolescent body image satisfaction scale for males: exploratory factor analysis and implications for strength and conditioning professionals. J. Strength Cond. Res. **28**(9), 2657–2668 (2014)
20. IPIP, Preliminary IPIP Scales Measuring Constructs Similar to Those Included in Lee and Ashton's HEXACO Personality Inventory (2015). Accessed 2015. http://ipip.ori.org/newHEXACO_PI_key.htm#Anxiety

21. Witmer, B.G., Jerome, C.J., Singer, M.J.: The factor structure of the presence questionnaire. Presence **14**(3), 298–312 (2005)
22. Fox, J., Bailenson, J.N.: Virtual self-modeling: the effects of vicarious reinforcement and identification on exercise behaviors. Media Psychol. **12**(1), 1–25 (2009)
23. Feng, A., Rosenberg, E.S., Shapiro, A.: Just-in-time, viable, 3-D avatars from scans. Comput. Anim. Virtual Worlds **28**(3–4) (2017)

Short Papers

Impact of Tutor Errors on Student Engagement in a Dialog Based Intelligent Tutoring System

Shazia Afzal(✉), Vinay Shashidhar, Renuka Sindhgatta, and Bikram Sengupta

IBM Research, Bengaluru, India
{shaafzal,vinays16,renuka.sr,bsengupt}@in.ibm.com

Abstract. Accurate classification of learner responses is a critical component of dialog based tutoring systems (DBT). Errors in identifying the intent and context of responses can have cascading effects on the ongoing interaction thereby affecting the learning experience and outcome. In this paper we attempt to quantify the impact of Tutor misclassifications on student behavior by analyzing differences across our hypothesized conditions namely, no-misclassification vs. misclassification using various dialog metrics. We find that not only are there significant changes in behavior across the two groups but that Tutor errors related to misunderstanding of Intent - although fewer in occurrence, appear to have a higher impact than a misclassification of a valid student answer. We also see some evidence of the effectiveness of scaffolds like FITBs in sustaining dialog thereby mitigating the effects of a Tutor error.

1 Introduction

ITS offer an effective mode of instruction by virtue of their ability to provide adaptive learning through personalized scaffolding and formative feedback [6,7]. A special case of ITS is Dialog-based Tutoring (DBT) which is based on the socratic principle of cooperative dialogue (e.g. AutoTutor [5]). A conversation is triggered when the Tutor poses a question which typically leads to a series of dialog turns directed towards finer reasoning on relevant concepts. The goal is to scaffold knowledge and provide constructive remediation akin to expert one-one human tutoring. We have built a DBT that facilitates a conversational style assessment of learners mastery on domain knowledge. The DBT is currently live and has been accessed by over 200 students at the time of this study. As part of the ongoing iterative assessment of the DBT we analyze the conversational transcripts obtained during live interaction with students. In this paper we report some initial findings on student behavior related to the effect of errors and misclassifications by the DBT. Specifically, we explore the impact of Tutor accuracy on student engagement measured in terms of pre-defined dialog metrics signifying behavioral changes. We find that both the incidence as well as the type of errors have a significant effect on conversation completeness and discourse patterns. The immediate application of these results is to build a disengagement

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 267–273, 2018.
https://doi.org/10.1007/978-3-319-91464-0_26

prediction model using the relevant dialog metrics as features. This would help in timely identification of learning impasses so that appropriate interventions can be launched. In general, these findings quantify the impact of misunderstanding user utterances and are therefore of relevance to chat-bots used in other domains also.

Prior works have used time, performance and problem-specific features as well as mouse movements for automatic detection of off-task behavior, gaming or disengagement in ITS for example [1–3]. These have mostly used logs of student-tutor interactions synchronized with observation-based assessments as ground-truth for modeling and have shown promising results. This work differs on two fronts. Firstly, it attempts to determine the impact of Tutor accuracy on student behavior and is therefore novel in its purpose. Secondly, it helps in revealing high-level interpretable features to drive future efforts in automatic prediction of disengagement. Specifically, the effect of Tutor misclassifications is studied in terms of specific response and discourse patterns like typing speed, use of I don't knows (IDKs), request for Hints, etc. Acknowledging the limitations of current natural language techniques (NLP) techniques, this helps in identifying opportunities of intervention or remedial action to keep the learner engaged.

2 Description of the Tutor

Our DBT provides remediation to learners through natural language discourse. Systematic turn-taking engages learners in a conversation style assessment of their mastery on a topic. The Tutor tracks the learners progress during the course of interaction and launches appropriate interventions according to predefined dialog strategies. The main components of our DBT are shown in Fig. 1. The most significant part of the DBT is the Natural Language Response Classifier with two primary sub-components: Intent Classifier and Student Response Analyzer (SRA). The Intent Classifier identifies a student utterance as either a valid on-topic answer, a request for hint/help, a valid question, direct feedback to the Tutor or an out of context response. This level of response classification is crucial to the effective working of the Tutor as an error at this stage can have cascading effects on the entire dialog flow. Student response analysis (henceforth SRA) is the task of labeling student answers with categories that can help a dialog system to generate appropriate and effective feedback on errors [4]. The

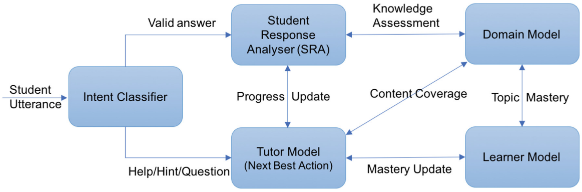


Fig. 1. Simplified architecture diagram

SRA takes the valid student answer and evaluates it against the model reference answer into one of 3 categories: correct, partial or incorrect. It also performs a gap analysis on the students answer against the expected one to generate fill in the blank (FITB) style prompts dynamically. The SRA in our DBT uses state of art machine learning techniques to perform classification with an accuracy of about 78%. The output of the Natural Language Response Classifier is used to drive the Tutor strategy by continuous evaluation against the domain model and estimates of mastery from the learner model. It is the importance of this unit to the overall functioning of the DBT that motivated us to experimentally analyze the effect of errors at this stage on student experience and behaviour. The DBT works in the typical ITS Outer and Inner Loop iterations to drive adaptive learning.

3 Study Design and Data

As the Tutor is built from limited training data its performance is susceptible to the open style natural language dialogue permitted in our DBT. Given the diversity of human language and technical limitations in achieving perfect language comprehension, NLP is still a challenging problem. In a DBT its accuracy determines the dialog flow and directly impacts learner engagement. This motivated us to study and quantify the impact of Tutor accuracy on student behavior. Our hypothesis was that misclassification of student utterance by the Tutor will result in decreased engagement that can be implicitly captured through dialog metrics and natural language cues.

We randomly selected 130 transcripts between the Tutor and Students for our study. Each transcript corresponds to a unique Tutor-student dialog session. Conversations having more than 4 turns were retained in order to filter out trials and inadequate learning sessions. This resulted in a sample set of 117 conversations with an average turn length of 9. For each conversation the dialog metrics listed in Table 1 were computed. Only 26 conversations had no Tutor error as compared to 91 that had at least one occurrence of Tutor misclassification. Table 1 shows a comparison of the average values for the metrics across the data splits. Though all metrics differ across the two sets of conversations overall, this comparison is not adequate because we cannot directly attribute this difference to Tutor error which is our aim. So to truly understand the impact of Tutor error we look at how the metrics vary within a conversational context. For this we exclude the conversations with zero misclassifications ($N = 26$) and take only the sample set with at least one Tutor error ($N = 91$) for the remainder of our analysis. We assume that a misclassification effect sets in when the student encounters a misclassification in the conversation. The turns following the Tutor error are supposed to be under its influence until an accurate classification takes the student back on track. This implies that the turns preceding the first Tutor error and the turns between a correct classification and an incorrect classification are not under misclassification effect and should have significantly different metric values. Our analyses confirms our assumptions. The important findings are discussed in the following section.

Table 1. Dialog metrics grouped by data splits

Metric *Avg.*	Description	All data *(N = 117)*	With error *(N = 91)*	No error *(N = 26)*
Turns	The number of student turns in a conversation	9.07	10.07	5.58
Tutor errors	Count of misclassifications in a conversation	2.32	2.98	0
Response time	Time taken by a student to answer	12.46	14.43	5.56
Response length	The number of characters in a response	340.84	378.33	209.62
Typing speed	Ratio of response length/response time	70.87	72.11	66.54
Help requests	Occurrence of IDKs (I don't know), IDUs (I don't understand), or similar help requests	0.39	0.36	0.5
ToTutor responses	Utterances that are explicitly targeted at the Tutor like 'I don't like you', 'You're wrong', 'I already said that', etc	0.33	0.43	0
Dialog Completion	Whether a conversation is completed or not	0.74	0.76	0.65
Fill in the blanks	No. of fill in the blanks (FITB) or prompts	0.88	0.96	0.62

4 Findings

Tutor Errors Cause Significant Changes in Behavior. Table 2 compares the metrics in the error and no-error conditions within a conversation. Considering each dialog metric as a dependent variable (DV) with respect to Tutor accuracy as independent variable (IV) we analyzed if the difference between metric values (DV) is significant when compared in no-misclassification versus misclassification (IV) conditions. As the distributions are not normal, we use the Wilcoxon signed-rank test to compare the differences across the metrics in the two conditions. We find that all metrics show statistically significant differences at $p < 0.01$. Specifically, the response length, response time and typing speed show a decreasing trend in the face of an error. This suggests that an erroneous classification by the Tutor impacts the manner in which students frame subsequent responses. One explanation for this could be that students take more time to think and formulate their answers resulting in a slower typing speed. A lower response length could be an indication of disinterest. Interestingly, the number of IDKs and To-Tutor responses both show a statistically significant increase in the misclassification condition. This could be attributed to the expression

Table 2. Comparison of dialog metrics in conversational context *($N = 91$)*

Metric	Tutor error *M, SD, Median*	No error *M, SD, Median*	Wilcoxon signed rank test *Z, $p < 0.01$*
Response length	130.41, 244.4, 50	247.92, 232.5, 180	−4.69
Response time	4.52, 15.8, 0.94	9.91, 35.42, 2.58	−3.91
IDKs	0.24, 0.58, 0	0.12, 0.43, 0	−1.75
To-tutor responses	0.34, 1.31, 0	0.09, 0.28, 0	−2.17
Typing speed	64.11, 43.71, 60.23	83.29, 57.31, 68.52	−2.83

of disagreement with the Tutor, annoyance at getting a right answer wrong or general remarks on inability instigating an explicit feedback to the Tutor. Refer here to Table 1 where the number of To-Tutor responses in case of conversations having no error is nil. Considering that there is a significant positive correlation between number of IDKs and To-Tutor responses ($r = 0.2$, $p = 0.02$), an increase in the frequency of explicit feedback to the Tutor together with the number of IDKs can be considered as an indication of students disengagement to serve as a prompt for launching interventions.

Tolerance to Tutor Errors Depends on the Type and Timing of Misclassification. We approximate tolerance to Tutor error by looking at dialog completion rate with respect to occurrence of misclassification. We see a significant negative correlation $r = -.24$, p $= 0.01$ implying that Tutor errors do impact the completion of a dialog. When considering the *type* of error, we find that the negative correlation is more pronounced for Intent errors ($r = -.31$, p $= 0.003$) than those of SRA ($r = -.22$, p $= 0.03$). To further explore the impact of type of error we consider the dialog completion rates versus the turn at which the error occurred. Specifically, we analyse whether a Tutor error in the initial few turns has an impact on student engagement. Considering that the average turn length of our sample set is 9 we look at the completion rates in four buckets corresponding to student turns less 2, between 2 and 5, between 5 and 9, and those greater than 9. Figure 2 shows how the dialog completion rate changes when sampled at these turn buckets. The trend shows that students are more tolerant to SRA errors occurring later on in the dialog as compared to early encounters with the same. This is an interesting finding as it seems to imply that the more time students have invested in the Tutor the more permissive they are to Tutor errors when it comes to understanding their answers. However, the opposite is true in the case of Intent misclassification. Although the results do not show significant differences the trend shows that the dialog completion rate decreases under the effect of Intent misclassification. This may imply that as the conversation progresses students are more sensitive to Intent misclassification and could be put off to the extent of exiting midway from the tutoring session. This is a very significant result, especially considering that the number of Intent classification errors are far less compared to the overall SRA errors. In

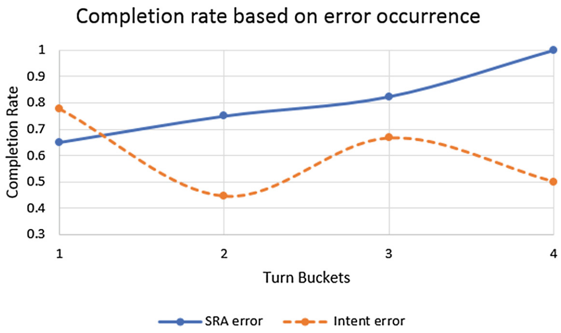


Fig. 2. Conversation completion rate with respect to turn

our original sample set of 117 conversations, there were a total of 215 response classification errors compared to only 56 intent classification errors.

Scaffolds Like Fill in the Blanks Improve Student Engagement. We find that the dialog completion rate is significantly higher when a student encounters a scaffolding prompt like a FITB (82%) as against a sharp correct or incorrect misclassification (61%), t(111.3) = 2.14, p = 0.03. There is also a significant positive correlation between the dialog completion rate and number of FITBs, $r = 0.22$, p = *0.03*. We analyse this further by distinguishing between incorrect FITB generation versus correct FITB generation. A correct FITB is generated when the Tutor rightly classifies a student response as partially correct whereas an erroneous FITB can be triggered because of Tutor misclassifying a response as partial. We observe that the completion rate on encountering at least one correctly generated FITB is significantly higher (M = 0.96) than when no FITB is generated following a Tutor misclassification (M = 0.68), t(65.48) = 2.15, p = 0.03. The same effect is not observed in case of an erroneous FITB.

5 Conclusions

Natural language classification is a critical component of conversational systems as understanding student intents and correctly scoring valid student responses is the very basis for driving an effective tutorial strategy. Given the limitations of NLP techniques it would be beneficial to implement interventions or scaffolds to alleviate the effects of Tutor misclassification of student utterances. To enable this we need to first understand how Tutor errors impact the conversational behavior of students so that these can be formalized as features for prediction modeling. This paper describes our analyses to do precisely this confirming our hypothesis that there are significant differences in student behavior based on the incidence and type of misclassifications they encounter. Going forward we aim to use a wider sample set to do a more fine-grained analysis and build a disengagement prediction model for just-in-time remediation.

References

1. Baker, R.S.: Modeling and understanding students off-task behavior in intelligent tutoring systems. In: Proceedings of SIGCHI Conference Human Factors, Computing Systems, pp. 1059–1068 (2007)
2. Beck, J.: Engagement tracing: using response times to model student disengagement. In: Proceedings of 12th International Conference Artificial Intelligence in Education (AIED 2005), pp. 88–95 (2005)
3. Cetintas, S., Si, L., Xin, Y.P., Hord, C.: Automatic detection of off-task behaviors in intelligent tutoring systems with machine learning techniques. IEEE Trans. Learn. Technol. **3**(3), 228–236 (2010)
4. Dzikovska, M.O., et al.: SemEval-2013 task 7: the joint student response analysis and 8th recognizing textual entailment challenge. In: The First Joint Conference on Lexical and Computational Semantics, SEM 2013, Atlanta, Georgia, USA, 13–14 June. Association for Computational Linguistics (2013)
5. Graesser, A.C., Lu, S., Jackson, G.T., Mitchell, H.H., Ventura, M., Olney, A., Louwerse, M.M.: AutoTutor: a tutor with dialogue in natural language. Behav. Res. Methods Instrum. Comput. **36**(2), 180–192 (2004)
6. Ma, W., Adesope, O.O., Nesbit, J.C., Liu, Q.: Intelligent tutoring systems and learning outcomes: a meta-analysis. J. Educ. Psychol. **106**(4), 901–918 (2014)
7. VanLehn, K.: The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. Educ. Psychol. **46**, 197–221 (2011)

Enhancing the Clustering of Student Performance Using the Variation in Confidence

Ani Aghababyan[1](✉), Nicholas Lewkow[1], and Ryan S. Baker[2]

[1] McGraw-Hill Education, New York City, USA
anie.aghababyan@gmail.com
[2] University of Pennsylvania, Philadelphia, USA

Abstract. While prior research has typically treated student self-confidence as a static measure, confidence is not identical in all situations. We study the degree to which confidence varies over time using entropy, investigating whether high variation in confidence is more characteristic of highly confident or highly uncertain students, using data from 118,000 students working within 8 courses within the LearnSmart adaptive platform. We find that more confident students are also more consistent in their confidence. Confident students were more likely to answer correctly but also more likely to be overconfident, making unexpected mistakes. Finally, we develop interpretable clusters of students based on their confidence entropy, degree of over/underconfidence, and related variables.

Keywords: Confidence · Variance · Entropy · Adaptive learning · Clustering

1 Introduction

As academic work becomes more and more reliant on remote or out of classroom participation, being able to account for learner characteristics that can affect their motivation and performance becomes vital. One of such motivational variables is one's self-belief expressed as self-confidence. As defined in research, confidence refers to one's beliefs in oneself and one's perceived abilities to succeed in a specific activity. Confidence refers to the strength of one's belief or the degree of confidence in a judgement.

Considerable research has shown connections between confidence and knowledge and has shown that confidence influences academic performance and outcomes [6]. Prior research has typically treated confidence as static, looking at overall levels of confidence, or confidence measured at a single time point. However, confidence is not identical in all situations, even for a given topic. Instead, it may be warranted to study the degree to which confidence varies over time and understand how variation in confidence relates to its overall levels. One possible way to represent how values vary is standard deviation, but this metric is poor at handling high variation and non-normal data. Other ITS researchers have used dynamic analyses to capture the variance of different student characteristics across contexts [9], but have not yet applied this method to study variation in student confidence over time. In this study, we investigate whether studying confidence entropy can enhance understanding of student performance. Confidence entropy could be beneficial to analysis of ITS in several ways, including the analysis of how it

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 274–279, 2018.
https://doi.org/10.1007/978-3-319-91464-0_27

relates to learning and performance, and also through incorporating it into clusters of students that can be used to differentiate learning experiences for different groups of students. As such, this paper will investigate whether confidence entropy can be a meaningful contributor to a successful and predictive set of clusters.

Thus, the goal of this paper is to better understand student confidence entropy and how it can contribute to enhancing clustering of students into meaningful groups. More specifically we plan to investigate the following:

- Is there variance in student confidence (confidence entropy) reports or do students generally experience and report consistent confidence levels over time? How does this variance correlate to students' average confidence level and to performance more broadly?
- Does confidence entropy meaningfully contribute to student clustering based on performance?

We hypothesize that student confidence entropy will contribute to a better-quality set of clusters that has better goodness metrics and can better predict student accuracy.

2 Data Set and Content

Our data comes from the LearnSmart adaptive platform that offers personalized learning and self-assessment adaptive paths. The platform provides immediate feedback on the accuracy of each answer along with an explanation of the correct answer. If the learners understand the content and are able to demonstrate knowledge, they progress quickly. If the learners are lacking knowledge, they will need to spend more time working through the questions. Since the courses we studied did not have a final grade within the platform, we used students' overall accuracy score instead, which is the ratio of student's correctly answered questions to their total number of questions answered.

LearnSmart measures student confidence by asking the learner to self-report their confidence after each question. Immediate ratings of confidence are used to reduce the frequency of inaccurate responses as a result of recall bias due to retrospection [5]. With each question, the platform prompts the student to select one of the confidence buttons from a four-level confidence scale: “I know it”, “Think so”, “Unsure”, “No Idea”. The system records these reports as “3”, “2”, “1”, “0” respectively.

For this study, we harvested data from eight courses from the Spring 2015 academic semester. We selected four humanities/social science courses and four physical/life science courses both with the largest usage. Additionally, we verified that the selected courses were comparable in terms of the number of total questions answered throughout the semester. Hence, the participants in the current study included 118,291 college students who took one of the eight courses taught via LearnSmart. Combined, these students completed 93,800,984 million questions.

3 Analysis 1: Confidence Entropy

Our first analysis attempts to better understand the variation of confidence, operationalized as Shannon entropy to find the distribution of confidence across its possible values

[3]. The Shannon entropy equation provides a way to estimate the average minimum number of bits needed to encode a string of symbols, based on the frequency of the symbols. The entropy index is calculated by the following formula:

$$h(p_1, \ldots, p_a) = -\sum_{i=1}^{a} p_i \log(p_i)$$

When entropy is zero, the learner's confidence never varies. If the entropy is the maximum value (2 in our case – the base-2 logarithm of the four possible outcomes), the learner used the four confidence levels in the same proportion; there is maximal uncertainty as to the student's confidence. In other words, higher entropy means higher variability in confidence reported, and lower entropy indicates consistency in the learner's confidence. Note that entropy calculation does not consider order.

In LearnSmart, the average confidence entropy was 0.78, suggesting that students' self-reported confidence does not vary much. It was also more common for a student to have very low entropy (0.1 or lower), 8.5% of students, than very high entropy (1.8 or higher). Only 2% of students have exactly 0 entropy. Of those 2% of students, 93% reported the highest confidence for every question, just under 7% reported the middle two confidence levels, and only 3 reported the lowest confidence. Students with 0 entropy also had a higher average accuracy than those with entropy above 0.

Across the distribution of students, a student's confidence entropy correlated to several other metrics. More confident students varied less in their self-reports: Confidence entropy and average confidence were correlated at r = −0.66. The majority of the learners who report only one level of confidence are also the learners who report high confidence. There is, however, a second group of low-entropy students who have an average confidence in the middle. Relatedly, students who varied less in their self-report were more likely to answer correctly; there was s a negative correlation between confidence entropy and student accuracy (r = −0.35). Previous research [1] found that learners with higher accuracy are also likely to have higher average confidence, as well as a higher proportion of overconfidence.

4 Analysis 2: Confidence Entropy

4.1 K-means Clustering Method

In our second analysis, we investigate whether students separate into relatively distinct groups based on their confidence entropy and other relevant performance characteristics captured by our set of variables. Thus, we use clustering analysis to build groups from the set of variables described below without including student accuracy, as we will correlate the clusters to this metric afterwards. We engineered performance features for each learner as input for this analysis. We chose the following features that are descriptive of a learner's performance but not dependent on their accuracy score:

1. Confidence entropy - variation of confidence described in analysis 1.

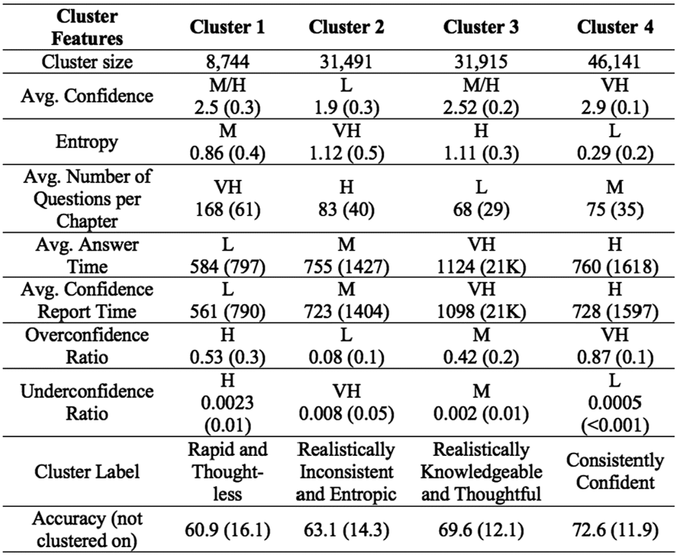
2. Overconfidence ratio - the proportion of incorrectly answered questions where the student reported the highest confidence.
3. Underconfidence ratio - the proportion of correctly answered questions where the student reported the lowest confidence.
4. Average confidence – mean confidence values for each student
5. Average number of questions answered - depending on the accuracy of their answer, each student may see 1 or more questions per learning objective
6. Average time taken to respond to the question (in 1/100ths of a second)
7. Average time taken to report confidence (in 1/100ths of a second) after prompt is displayed (after the student answers the question)

When using a clustering approach, several issues must be considered: the selection of clustering algorithm, the number of clusters, the statistical difference between clusters, cluster stability, and the interpretation of the clusters. Prior to including #5 as a feature, we verified that it was not a proxy for student accuracy ($r = -0.16$). To group the students into clusters we used the K-means clustering algorithm, which partitions the input into k distinct groups based on cluster centroid locations. We compared cluster consistency of the k-means to hierarchical clustering using silhouette validation [2] and k-means outperformed hierarchical clustering. Additionally, we used one-way ANOVA to compare the cluster mean for each feature in each cluster to make sure the average values of each cluster's features are significantly different from each other to render meaningfully different groups. Finally, for interpretation we came up with descriptive labels for each cluster and computed the average accuracy scores for each cluster to see whether the scores matched with our interpretation of the cluster performance based on the cluster characteristics.

4.2 K-means Cluster Results

Our cluster features had different scales, so prior to using the k-means algorithm, we converted them to z-scores. We used within-set sum of squared error between points in clusters to choose our cluster number. As a result, 4 was the highest number of clusters where within-set sum of squared errors was decreasing substantially. It also has reasonably-sized clusters. We then conducted one-way ANOVA. All 4 of our clusters are significantly different from each other on all seven of the features (all feature means had $p < 0.01$). Given that the probability of this pattern being obtained by chance is 0.01^{42}, further post-hoc correction is not needed. Finally, in order to enhance interpretability, we assigned low, medium, high, very high to the average values for each feature within each cluster based on their value (see Table 1).

Table 1. Mean and (standard deviation) of features for each cluster.



Cluster Features	Cluster 1	Cluster 2	Cluster 3	Cluster 4
Cluster size	8,744	31,491	31,915	46,141
Avg. Confidence	M/H 2.5 (0.3)	L 1.9 (0.3)	M/H 2.52 (0.2)	VH 2.9 (0.1)
Entropy	M 0.86 (0.4)	VH 1.12 (0.5)	H 1.11 (0.3)	L 0.29 (0.2)
Avg. Number of Questions per Chapter	VH 168 (61)	H 83 (40)	L 68 (29)	M 75 (35)
Avg. Answer Time	L 584 (797)	M 755 (1427)	VH 1124 (21K)	H 760 (1618)
Avg. Confidence Report Time	L 561 (790)	M 723 (1404)	VH 1098 (21K)	H 728 (1597)
Overconfidence Ratio	H 0.53 (0.3)	L 0.08 (0.1)	M 0.42 (0.2)	VH 0.87 (0.1)
Underconfidence Ratio	H 0.0023 (0.01)	VH 0.008 (0.05)	M 0.002 (0.01)	L 0.0005 (<0.001)
Cluster Label	Rapid and Thought-less	Realistically Inconsistent and Entropic	Realistically Knowledgeable and Thoughtful	Consistently Confident
Accuracy (not clustered on)	60.9 (16.1)	63.1 (14.3)	69.6 (12.1)	72.6 (11.9)

Based on Table 1, Cluster 1 students ("Rapid & Thoughtless") show signs of low effort, both on answering questions and reflecting on their confidence, spending the lowest amount of time thinking about the questions. These students do not appear to be realistic in their expectations as they have one of the highest average confidence scores, despite being required to complete double as many items as other students due to making many errors, and had the second-highest overconfidence ratio. Cluster 2 students ("Realistically Inconsistent/Entropic") have the lowest average confidence among all the Clusters, are least likely to be overconfident, and most likely to be underconfident (although underconfidence was still rare). In addition, these students have the highest entropy, varying considerably in their answers about their confidence, and using the middle confidence buttons more often than the extremes. Cluster 2 students spend an adequate amount of time answering questions but make many errors and have to answer more questions than average. Students in Cluster 3 ("Realistically Knowledgeable and Thoughtful") completed chapters with the fewest number of questions of any cluster but spent the longest responding to questions, suggesting that these students put extra effort into their work. These students are confident, but unlike Cluster 1, have high entropy in their reports of their confidence. Finally, Cluster 4 students ("Consistently Confident") have the highest confidence and the lowest entropy, mostly choosing the highest confidence button. On average these students answer a relatively small number of questions per chapter, due to successful performance. These students spend a moderate amount of time answering questions. However, these students had the highest overconfidence ratio by a substantial amount.

After labeling the groups, we correlated cluster membership to students' actual accuracy. As Table 1 shows, Cluster 4, Consistently Confident, had the highest mean accuracy, 72.6 (SD = 11.88), a finding in line with their very high average confidence, low entropy, and a very high overconfidence ratio (overconfidence has been found to be associated with good academic performance [1]). Cluster 1, Rapid & Thoughtless, was the lowest performing group, unsurprising given their low average time spent per question and very high average number of questions answered per chapter. Clusters 2 and 3 were in the middle. These findings suggest that our clustering approach including entropy led to a meaningful and interpretable set of clusters that corresponded closely to student accuracy, despite not having actual accuracy information to cluster on.

5 Discussion and Future Work

In this paper, we explored students' confidence variability, operationalizing this as confidence entropy. We then examined the variability of students' self-confidence, and analyzed its relationship with performance and confidence strength. Our results show that average confidence and confidence entropy are highly negatively correlated, suggesting that more confident students are also more consistent in their confidence. More consistent confidence is also associated with higher accuracy. We then developed meaningful, interpretable clusters using entropy in combination with other behavior variables. Confidence and confidence entropy could be used in several ways in future ITS research and practice, including using time-based confidence entropy to predict if a student is losing interest or changing their outcome expectations. The clusters developed here could also be used to provide differential learning experiences. Future work should take context into account as well, investigating if some students' confidence varies more in specific situations or for specific material. As such, the work here is only a step towards better understanding how confidence shifts over time, and how this understanding can be used to improve learning.

References

1. Aghababyan, A., Lewkow, N., Baker, R.: Exploring the asymmetry of metacognition. In: Proceedings of the Seventh International Learning Analytics and Knowledge Conference, pp. 115–119. ACM, New York (2017)
2. Rousseeuw, P.J.: Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J. Comput. Appl. Math. **20**, 53–65 (1987)
3. Shannon, C.E.: Prediction and entropy of printed English. Bell Labs Tech. J. **30**(1), 50–64 (1951)
4. Snow, E., Jacovina, M., Varner, L., Dai, J., McNamara, D.: 2014. Entropy: a stealth measure of agency in learning environments. In: Proceedings of the 7th International Conference on Educational Data Mining, pp. 241–244, Springer, Heidelberg (2014)
5. Stone, A., Shiffman, S., Atienza, A., Nebeling, L.: The Science of Real-Time Data Capture: Self-Reports in Health Research. Oxford University Press, Oxford (2007)
6. Zusho, A., Pintrich, P.R., Coppola, B.: Skill and will: the role of motivation and cognition in the learning of college chemistry. Int. J. Sci. Educ. **25**(9), 1081–1094 (2003)

Intelligent Virtual Reality Tutoring System Supporting Open Educational Resource Access

Jae-wook Ahn[1](✉), Ravi Tejwani[1], Sharad Sundararajan[1], Aldis Sipolins[1], Sean O'Hara[1], Anand Paul[1], Ravi Kokku[1], Jan Kjallstrom[2], Nam Hai Dang[2], and Yazhou Huang[2]

[1] IBM T.J. Watson Research Center, Yorktown Heights, USA
{jaewook.ahn,rtejwan,sharads,asipoli,sean.ohara,
anand.paul,rkokku}@us.ibm.com
[2] EON Reality, Inc., Irvine, USA
{jan,namhai,david.huang}@eonreality.com

Abstract. Virtual Reality is gathering increasing popularity for Intelligent Tutoring Systems. We introduce an approach that improves the baseline VR experience for ITS by enabling access to open educational resources and more intelligent navigation with the support of multiple artificial intelligence algorithms. A preliminary user study result not only reveals the potential of the proposed method, but also helps to identify the clues to improve the current design.

Keywords: Virtual reality · Intelligent Tutoring System
Open educational resources · Anatomic structure education

1 Introduction

There are a growing number of research publications in the recent years on using Virtual Reality for Education [8,9]. Virtual reality (VR) refers to technology that presents a three-dimensional environment via a head-mounted display. The display presents a slightly different image to each eye, creating an effect called binocular disparity that matches how we perceive the real world. The result is a convincing feeling of immersion, also called presence, that makes virtual environments feel real. By contrast, a virtual environment displayed on a conventional display is perceived as a flat image because it lacks binocular disparity.

For Intelligent Tutoring Systems (ITS), VR provides students with immersive experience, so that they can focus more on learning goals and gain knowledge that is hard to be learned from traditional training or from simple 2D visualizations. Among the various possibilities that VR ITS could provide, anatomic structure education has been one of the popular domains, e.g., [11]. It is due to the fact that the domain requires multi-dimensional representation of the objects (i.e., anatomic organs) with the ability to dissect those objects to better learn

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 280–286, 2018.
https://doi.org/10.1007/978-3-319-91464-0_28

about them [5]. A large amount of the information it should deliver to the students also fits well with the immersive nature of the approach. However, we also feel that simple utilization of the 3D VR technology is insufficient to efficiently educate the required knowledge, considering the amount and the diversity of the learning materials.

In this paper, we introduce an approach to promote the use of VR in biology education, by linking the VR technology with various artificial intelligence and cognitive technologies such as exploratory search [7], recommendation, and adaptive navigation to open educational resources (OER) [3]. Several related approaches have been introduced for adaptive learning and hypermedia. For example, Chittaro and Ranon's approach [4] is very similar to ours. They discuss the adaptivity in the context of 3D Web sites and with respect to Web-based hypermedia, disassembling a particular object based on the ontology and providing educational resources to the learner. However, it is not an immersive VR representation as introduced in this paper. Brusilovsky [2] describe *adaptive navigation support* technologies that support user navigation in hyperspace, by adapting to the goals, preferences, and knowledge of the individual user. Kaufmann et al. [6] describe a 3D geometry construction tool specifically for mathematics and geometry education, based on mobile collaborative Augmented Reality. It promotes and supports dynamic exploratory behavior.

Inspired by these previous endeavors, we propose a novel VR Intelligent Tutoring System that supports direct manipulation and exploratory navigation of anatomic sub-structures and accessing open educational resources with the help of various artificial intelligence techniques. The adoption of those technologies on top of the VR interaction layer is critical in order to provide a better learning experience through more powerful search and navigation towards rich resources. We also provide a preliminary use case and user feedback results.

2 System Design and Implementation

We have designed a prototype system following the flow in Fig. 1. A learner interacts with a 3D frog model in a VR space. The model is comprised of hierarchical parts of a frog, which are viewed and manipulable from all directions and selected by a leaner using a pointer. An avatar provides verbal descriptions using Text-to-Speech (TTS) about a selected part so that the learner can learn about the characteristics of the part. A video content about the part is displayed on a virtual screen as well.

In addition to providing prescribed instructions about the frog organs, the system allows users to search for open educational resources and get recommendations from the system. It is known that users tend to provide too short queries (2 or 3 words) and not good at expressing their information needs with higher quality queries. It may be even more challenging to receive effective user queries in a VR environment where users' expressive capacity is limited than conventional web environment. Therefore, a mechanism to increase the search power of original queries is required by expanding them [12]. We employed an amphibian

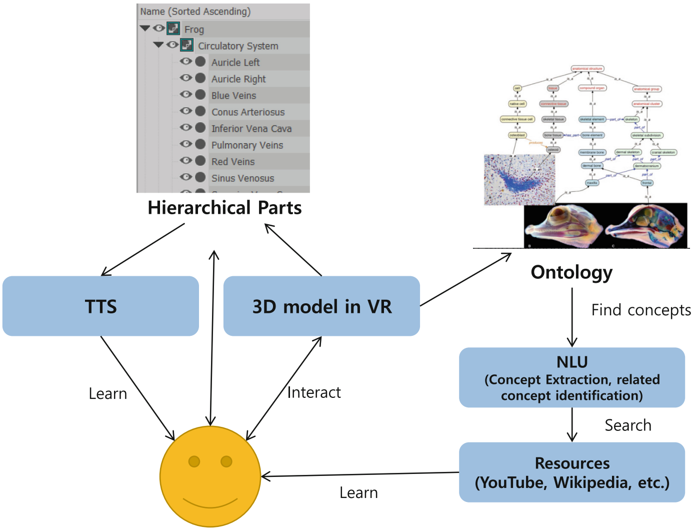


Fig. 1. VR ITS system flow. A learner interacts with a 3D model in VR and learns about a frog part through Text-To-Speech. She requests for related open resources about the part or recommendation for other related parts. The task is done by looking up the ontology, extracting concepts, and running search against online databases.

ontology (Amphibian Gross Anatomy Ontology (https://bioportal.bioontology.org/ontologies/AAO) [10]) and IBM Watson Natural Language Understanding (NLU) service [1]. Using the ontology, additional cues instead of a simple part name can be achieved. The ontology we used provides definitions of a specific amphibian part and information about the hierarchical relationships of it. The NLU module we used can extract concepts, keywords, and entities from any given texts, so we could select better query terms from the ontology information by avoiding noises. The expanded and refined queries are run against internal or third-party search engines and retrieves related resources. At the same time, related organs or parts can be recommended using the same information used for the search.

Figure 2 shows a screenshot of the prototype. The digital environment and interactions were created in the Unity3D (https://unity3d.com) game engine and rendered on a high-end desktop computer with a GTX 1080 graphics card. Users wore an HTC Vive (https://www.vive.com/us) headset to view the environment, which enables full rotation and position tracking in a room-scale (15'x15') space. Users interacted with objects by using wireless controllers included with the HTC Vive. The VR experience had users examine the skeletal and biological structure of a frog. A virtual avatar (Fig. 2(a)) guided the user through the experience by delivering verbal instruction via a cognitive Text-To-Speech service. Responses were pre-scripted and triggered by user interaction. For example, when the user is prompted to “look at the [frog's] right eye”, the next verbal guidance is delivered

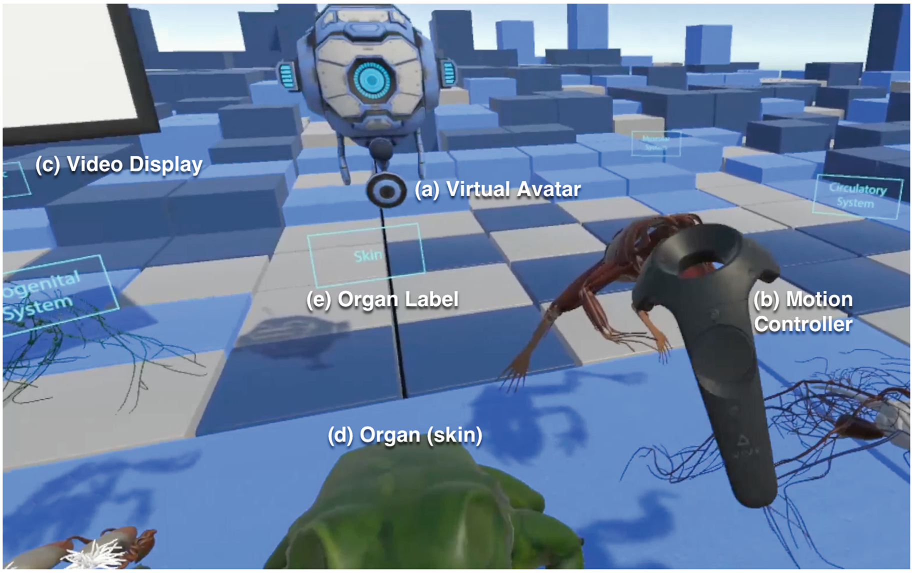


Fig. 2. VR ITS space screenshot. An anatomic structures of a frog is displayed in a 3D VR space. A learner explores the space, selects a part using a pointer, and learns about it. A Virtual Avatar plays a role of a tutor by providing verbal instructions or explanations.

only when the user rotates their headset in the direction of the relevant 3D model. Another interaction prompts the user to "select the skull" and proceeds only once the user has touched the frog skull with the motion controller (b). A virtual screen (c) floating in front of the user displayed supplemental video content to enrich the experience. Users could view a whole frog and then separate it into labeled (e) sub-structures (d). Users could also view a semi-transparent 'X-ray' version of the frog that made visible its biological substructures.

3 Prototype System User Study

We defined two research questions and conducted a preliminary user study about the prototype introduced in the previous section:

1. Is it easy to use the interactive elements (visual, voice, and the video recording) of the prototype while exploring the VR space?
2. Is it helpful to have access to adaptive navigation features such as OER resource search and related part recommendation?

For the study, twenty researchers were recruited within our institution. All the participants are experts in education technology research and development. In the first part of the user study, we presented a video recording showing the interaction use case of the prototype to the participants. The actual VR experience could not be provided due to the limited resources and the participants' locations but the user study was to test the visual aspects of the prototype and the lack of virtuality and the interactivity does not bias the results. It also ensures all participants having exactly the same experience to form their evaluations. In the second part, we asked each participant to try 3 queries against our frog part search facility that makes use of the AI technology-based query expansion and searching external resources. After the two trials, the participants were asked to answer a survey about the features of the prototype using a 5-point Likert Scale.

About the VR experience, the first group of the questionnaires asked about the effectiveness of the visual elements of the prototype: 3D frog parts, frog part labels, and the overall layout of the parts. The second group is about the audio related features such as voice instructions and explanations. The last group is about the OER access: related materials and navigation. Table 1 summarizes the results. Overall, the participants expressed positive experiences about the system (average score = 4.0 out of 5.0). They were satisfied with the quality of the TTS features (4.13 and 4.07) and gave high scores to the possibility of the OER resource search and the navigation feature (4.07 and 4.20). However, the visual elements needs improvements. Especially the layout of the frog parts recorded the lowest (2.93). This may be because the parts were scattered around the space for exploration but cluttered the space at the same time. A better approach needs to be developed that can adaptively locate, reduce clutter, and decide more efficient layout of the objects in the virtual space.

Table 1. User feedback on VR interaction

Visual				TTS		Resources		
Parts	Labels	Layout	Video	Instruction	Explanation	Related materials	Navigation	Overall
4.13	3.93	2.93	4.07	4.13	4.07	4.07	4.20	4.00

The second part that asked about the effectiveness of the query expansion and searching is less satisfactory (Table 2). We asked the participants to rate the quality of the expanded query and the search results. The scores are slightly above the neutral which reflects dissatisfaction from some of the participants. After examining the search log, we found that only 7 out of 35 queries entered by the participants were expanded using the ontology and the NLU. Several queries (part names) did not match with the ontology headings due to the difference of the languages and that lead to the failure to apply the NLU service and eventually may have affected the final search results.

Table 2. User feedback on query expansion and searching

Query expansion quality	Search quality using the expanded query
3.60	3.31

4 Conclusions

In this paper, we introduce a Virtual Reality based Intelligent Tutoring System that teaches amphibian anatomy. In addition to the conventional VR experience, we attempt to empower the VR with an ability to navigate and access open educational resources with the help of artificial intelligence technologies. A prototype was designed and implemented, and a preliminary user study was conducted to answer initial research questions. The participants of the user study provided important information on improving the current implementation as well as supporting the potential of our approach. We plan to enhance the visual layout of the learning objects within the VR space by developing adaptive layout methods within the VR space. We also plan to address the issues of our educational resource access algorithm discovered during the user study.

References

1. Watson Natural Language Understanding (2018). https://www.ibm.com/watson/services/natural-language-understanding/
2. Brusilovsky, P.: Adaptive navigation support. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) The Adaptive Web. LNCS, vol. 4321, pp. 263–290. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72079-9_8
3. Brusilovsky, P.: Adaptive navigation support for open corpus hypermedia systems. In: Nejdl, W., Kay, J., Pu, P., Herder, E. (eds.) AH 2008. LNCS, vol. 5149, pp. 6–8. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-70987-9_2
4. Chittaro, L., Ranon, R.: Adaptive 3D web sites. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) The Adaptive Web. LNCS, vol. 4321, pp. 433–462. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72079-9_14
5. Jang, S., Vitale, J.M., Jyung, R.W., Black, J.B.: Direct manipulation is better than passive viewing for learning anatomy in a three-dimensional virtual reality environment. Comput. Educ. **106**, 150–165 (2017)
6. Kaufmann, H., Schmalstieg, D., Wagner, M.: Construct3D: a virtual reality application for mathematics and geometry education. Educ. Inf. Technol. **5**(4), 263–276 (2000)
7. Marchionini, G.: Exploratory search: from finding to understanding. Commun. ACM **49**(4), 41–46 (2006)
8. Marks, S., White, D., Singh, M.: Getting up your nose: avirtual reality education tool for nasal cavity anatomy. In: SIGGRAPH Asia 2017 Symposium on Education, SA 2017, pp. 1:1–1:7. ACM, New York (2017)
9. Merchant, Z., Goetz, E.T., Cifuentes, L., Keeney-Kennicutt, W., Davis, T.J.: Effectiveness of virtual reality-based instruction on students' learning outcomes in K-12 and higher education: a meta-analysis. Comput. Educ. **70**, 29–40 (2014)

10. Musen, M.A., Noy, N.F., Shah, N.H., Whetzel, P.L., Chute, C.G., Story, M.-A., Smith, B., NCBO Team: The national center for biomedical ontology. J. Am. Med. Inf. Assoc. **19**(2), 190–195 (2011)
11. Seo, J.H., Smith, B.M., Cook, M., Malone, E., Pine, M., Leal, S., Bai, Z., Suh, J.: Anatomy builder VR: applying a constructive learning method in the virtual reality canine skeletal system. In: Andre, T. (ed.) AHFE 2017. AISC, vol. 596, pp. 245–252. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-60018-5_24
12. Xu, J., Croft, W.B.: Query expansion using local and global document analysis. In: Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 1996, pp. 4–11. ACM, New York (1996)

Emotional State and Behavior Analysis in a Virtual Reality Environment: A Medical Application

Hamdi Ben Abdessalem(✉), Marwa Boukadida, and Claude Frasson

Département d'Informatique et de Recherche Opérationnelle,
Université de Montréal, Montréal, H3C 3J7, Canada
{benabdeh,boukadim,frasson}@iro.umontreal.ca

Abstract. The performance and reactions of an individual in urgent situations vary according to his emotional states, which are subject to sudden changes. In medical situation, when patient's life is in danger, the doctor's emotional states are provoked and could affect his decisions. Virtual reality environments represent immersive situations in which we can dynamically simulate medical cases and particularly emergency cases. Using virtual reality and EEG devices, we can analyze the doctor's emotional state and behavior without risk. In this paper, we propose to generate medical cases that can induce frustration or stress which can be at the origin of mistakes from the student. For that, we created a neurofeedback system named "Hypocrates" composed of a virtual reality environment, a medical cases generator and an intelligent agent. An experimental study involving 15 students in medicine was conducted to evaluate our approach. Results show that, the mistakes generally increase the frustration of medical students and decrease their performance.

Keywords: Virtual reality · Neurofeedback · Intelligent agent
Emotional intelligence · EEG · Simulation-based learning · Medicine

1 Introduction

The emotional state of human beings changes regularly and sometimes suddenly. In order to study the changes of emotional states, it is better to interact with a virtual world in which we can modify several parameters freely and without constraints. Several researches focused on studying and analyzing the emotional states of human beings in the fields of advertising [1], video games [2], etc. but few researches focused on analyzing the emotional states in the field of medicine, despite the challenges and risks of this field.

Physiological measurements, like cerebral activity (EEG), electrodermal activity (EDA) and eye tracking allow a better understanding of the user's emotional reactions while interacting with the environment. Virtual reality environments can generate extreme emotions due to the sense of presence and, therefore, physiological sensors are indispensable in order to study and analyze the evolution of users' emotional state. In

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 287–293, 2018.
https://doi.org/10.1007/978-3-319-91464-0_29

addition to physiological measurements and virtual reality, neurofeedback helps further understanding of emotions because it gives a real-time feedback of what the user feels.

In this paper, we propose to analyze the behavior and reactions of medical students by analyzing their emotional states, before and after two types of events: (1) a sudden evolution of the patient that will cause stressful situations due to a new emotional state, and (2) a mistake of the student in the clinical reasoning.

Therefore, we combine virtual reality technologies with physiological measurements and neurofeedback. Our main hypothesis is the following: “Emotional situations can have an impact on physician’s decision-making abilities”.

The rest of this paper is organized as follows. An overview of the related work is given in Sect. 2. In Sect. 3, we present “Hypocrates” our neurofeedback system, which contains a medical cases generator and a neural agent in addition to the virtual reality environment. Section 4 details the experiment, which led to the results presented and discussed in the final section.

2 Related Work

2.1 Virtual Reality

Thanks to remarkable progress in the recent years, virtual reality started to be used in many fields. In fact, this technology which keeps on progressing every day, has a lot of advantages. The main advantage of virtual reality compared to other interactive environments is that the user is isolated from external visual distractions. The immersion can make the user believe that he is in a real world [3]. This technology has been applied in the field of psychology, to treat various disorders including brain damage [4], anxiety disorders [5] and alleviation of fear [6].

2.2 Neurofeedback

Neurofeedback is a type of biofeedback that measures brain waves to produce a signal that can be used as feedback. When measured activity is brain activity, biofeedback is called neurofeedback [7]. Neurofeedback has been used in the field of video games. Ben Abdessalem and Frasson [8] have proposed a neurofeedback approach to adapt video games to players according to their cognitive and emotional states. In their work, they proposed to follow and adapt in real-time the parameters of the video game according to the level of frustration and excitement of the players.

2.3 Brain Assessment

Most studies in the brain assessment field have relied on EEG signals to detect, analyze and evaluate emotions and mental states. Some researchers used EEG data in the detection of emotions for improving learning. In the field of emotions’ detection, there is a study that has been designed to detect the real-time valence (positive or negative emotion) of participants while they were watching videos [9]. Moreover, Ghali et al.

[10] have proposed to improve intuitive reasoning through help strategies in a virtual reality game. In fact, they assessed participants' mental states by considering their engagement and frustration in the game using EEG.

3 "Hypocrates": A Neurofeedback System

Our goal is to track in real-time the emotional state of medical students and intervene in the virtual environment in order to change their emotional state and analyze subsequently their reactions after mistakes in clinical reasoning or after interventions of an intelligent agent able to create a stressful situation. For this purpose, we created "Hypocrates", a neurofeedback system composed of three main parts: the **virtual reality environment**, the **Medical Cases Generator**, and the **neural agent** (see Fig. 1).

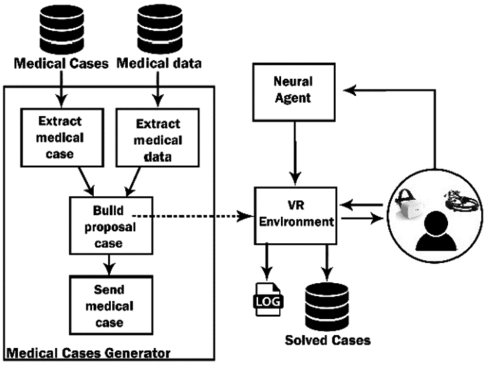


Fig. 1. Architecture of "Hypocrates"

"Hypocrates" contains three databases: (1) a medical cases database that contains different correct medical cases, (2) a medical data which contains extra medical data, (3) a solved cases database that contains the different cases solved by the students.

3.1 Virtual Reality Environment

In order to test our approach, we started by creating an interactive environment. This virtual reality environment is a dynamic system able to present various medical cases in real-time and produce realistic 3D objects and sound in order to be immersive. This environment is an important component in our system because it is what the student can constantly see. In this environment, the medical student is immersed in several scenes. He initially goes through an introductory scene in which we expose and explain how to interact with this virtual environment. Subsequently, he is exposed to a virtual operating room or a doctor's office, depending on the type of the medical case to solve. In each case, the medical student looks at the panel displayed in the environment which contains the symptoms of the patient and information, a reliability score gauge, an "Analysis" and a "Diagnosis" buttons (see Fig. 2). The medical student has to read the displayed symptoms, then asks for analysis if needed (a panel containing a list of analysis will appear) and the results of demanded analysis appear. Next, he selects a medical diagnosis. Once the choice of diagnosis is selected a series of panels appear, each one containing three possible actions, (one correct, and two wrong). The number of these panels depends on the number of actions to perform in the current medical case. The participant interacts with the virtual environment through a virtual reality headset and a gamepad.

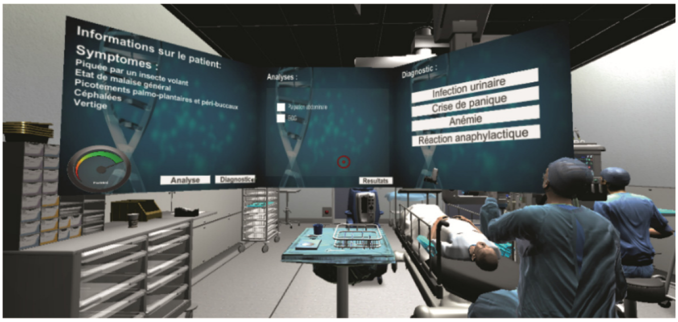


Fig. 2. Example of a problem case

3.2 Medical Cases Generator

This component handles the coordination between the two databases and the virtual environment. The role of the medical cases generator is to generate a problem case to be submitted to the student. Therefore, it combines each medical case with extra medical data to generate the **problem case**. The goal is to produce a case with correct and wrong data so that the student should select only the correct data. Then, it sends this problem case to the virtual environment in order to be exposed to the student. Figure 2 illustrates the problem case generated by the medical cases generator and displayed in the virtual reality environment.

3.3 Neural Agent

The neural agent is an intelligent agent that tracks the emotional state of the student and intervenes on the virtual reality environment in order to modify the student's emotional state. It uses a measuring module, which handles the detection of different mental states and emotions through EEG capture [8]. The neural agent modifies the problem case in a way to provoke the student, for instance, adding "internal bleeding" as a new symptom (which should provoke student's stress). After that, the agent analyzes the result of the intervention and could intervene again on the virtual reality environment in order to adapt it to the user.

4 Experiment

In order to study the effectiveness of our approach we experimented our system on 15 participants (8 males and 7 females, mean age 25.66, SD age = 4.31). The goal of this experiment is to confirm our research hypothesis using "Hypocrates". Our strategy is to use "Hypocrates" and intervene in the virtual reality environment in order to increase frustration of the medical students. We aim to check the reliability of the student's decision in emotional situations. The experimental protocol is the following. In the first step of the experiment, the participant signs an ethic form which explains the study and fills a pre-session form. In the second step, we install an Emotiv EPOC headset. In the third step, the participant is equipped with the FOVE VR headset and we give him a wireless gamepad to interact with the environment. After these steps, we start the "Medical Cases Generator", the Neural Agent and the virtual reality environment.

5 Results and Discussion

To confirm the effectiveness of our approach and to study the emotional reactions of the participants, we analyzed their frustration before and after they made a mistake. In fact, we calculated the average of frustration, 5 s before the mistake and the average of frustration 5 s after the mistake.

We conducted a paired-samples t-test to compare the frustration of the medical student before and after the mistake. We note that participants made 180 mistakes. As shown in Table 1, result shows that the average frustration after the mistake compared to the average frustration before the mistake went from 0.441 to 0.551, t(179) = 11.0075 and p = 0.000 * < 0.01. This result is significant and we can confirm that the average frustration state of medical students after the mistake is greater than the average frustration before the mistake by 11%. Therefore, when the participants made a mistake their level of frustration increases and that could affect their decisions. Thus, it is very interesting to analyze the impact of the increased frustration on their decisions.

Table 1. T-test results (before and after a mistake)

	Before a mistake	After a mistake
Mean (frustration)	0.4419	0.5513
SD	0.1662	0.1951
N	180	180
T	11.0075	
P	0.0000*	

As explained in the methodology, the neural agent intervenes in the virtual reality environment to provoke the medical student, so we compared the wrong decisions of the participants before and after these interventions. An example of intervention consist on adding new serious symptoms like "internal bleeding" in addition to sound effects, so the student should react quickly to save the patient. Results show that, before the agent's intervention, the average of the successive wrong decisions of the 15 participants is 2.13, whereas, after the intervention of the agent and the increase in the level of frustration, the average of the successive wrong decisions becomes 3.67. These results show that the performance of medical students can decrease with the increase of frustration.

6 Conclusion

In this paper, we presented "Hypocrates", a neurofeedback system in order to analyze the emotional state and behavior of medical students in reasoning situations. For that, we created a virtual reality environment to dynamically present medical cases able to be remotely modified by the neural agent. We conducted experiments and tests. Results showed that, on one side, the mistake of medical students affects their level of frustration, and on the other side, the level of frustration affects the decision-making process of the students and can lead to mistakes. Further work will aim to learn the behaviors and reactions of the students using machine learning techniques and our database of solved cases in order to predict actions and warn the students in advance for possible mistakes.

Acknowledgments. We acknowledge NSERC-CRD and Beam Me UP for funding this work.

References

1. Venkatraman, V., Dimoka, A., Pavlou, P.A., Vo, K., Hampton, W., Bollinger, B., Hershfield, H.E., Ishihara, M., Winer, R.S.: Predicting advertising success beyond traditional measures: new insights from neurophysiological methods and market response modeling. J. Mark. Res. **52**, 436–452 (2015)
2. Chen, D., James, J., Bao, F.S., Ling, C., Fan, T.: Relationship between video game events and player emotion based on EEG. In: Kurosu, M. (ed.) HCI 2016. LNCS, vol. 9733, pp. 377–384. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39513-5_35
3. Biocca, F.: The cyborg's dilemma: progressive embodiment in virtual environments [1]. J. Comput.-Mediat. Commun. **3** (2006)
4. Rose, F.D., Brooks, B.M., Rizzo, A.A.: Virtual reality in brain damage rehabilitation: review. Cyberpsychol. Behav. **8**, 241–262 (2005)
5. Gorini, A., Riva, G.: Virtual reality in anxiety disorders: the past and the future. Expert Rev. Neurother. **8**, 215–233 (2008)
6. Alvarez, R.P., Johnson, L., Grillon, C.: Contextual-specificity of short-delay extinction in humans: renewal of fear-potentiated startle in a virtual environment (2007)
7. Sherlin, L.H., Arns, M., Lubar, J., Heinrich, H., Kerson, C., Strehl, U., Sterman, M.B.: Neurofeedback and basic learning theory: implications for research and practice. J. Neurother. **15**, 292–304 (2011)
8. Ben Abdessalem, H., Frasson, C.: Real-time brain assessment for adaptive virtual reality game: a neurofeedback approach. In: Frasson, C., Kostopoulos, G. (eds.) Brain Function Assessment in Learning. LNCS (LNAI), vol. 10512, pp. 133–143. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-67615-9_12

9. Soleymani, M., Asghari-Esfeden, S., Pantic, M., Fu, Y.: Continuous emotion detection using EEG signals and facial expressions (2014)
10. Ghali, R., Ben Abdessalem, H., Frasson, C.: Improving intuitive reasoning through assistance strategies in a virtual reality game. In: The Thirtieth International Florida Artificial Intelligence Research Society Conference. AAAI, Florida (2017)

Deep Learning in Automated Essay Scoring

David Boulanger(✉) and Vivekanandan Kumar

Athabasca University, Edmonton, AB T5J 3S8, Canada
{dboulanger,vivek}@athabascau.ca

Abstract. This paper explores the application of deep learning in automated essay scoring (AES). It uses the essay dataset #8 from the Automated Student Assessment Prize competition, hosted by the Kaggle platform, and a state-of-the-art Suite of Automatic Linguistic Analysis Tools (SALAT) to extract 1,463 writing features. A non-linear regressor deep neural network is trained to predict holistic scores on a scale of 10–60. This study shows that deep learning holds the promise to improve significantly the accuracy of AES systems, but that the current dataset and most essay datasets fall short of providing them with enough expertise (hand-graded essays) to exploit that potential. After the tuning of different sets of hyperparameters, the results show that the levels of agreement, as measured by the quadratic weighted kappa metric, obtained on the training, validation, and testing sets are 0.84, 0.63, and 0.58, respectively, while an ensemble (bagging) produced a kappa value of 0.80 on the testing set. Finally, this paper upholds that more than 1,000 hand-graded essays per writing construct would be necessary to adequately train the predictive student models on automated essay scoring, provided that all score categories are equally or fairly represented in the sample dataset.

Keywords: Deep learning · Automated essay scoring · Writing analytics

1 Introduction

Monitoring the writing process of students and offering valuable formative feedback, in real time, to students in need has long been an aspiration of teachers. Given that the linguistic analysis of any piece of writing, both formative and summative, relies heavily on its writing construct (the writing context), automated essay scoring requires deeper understanding and application of both data science and English teaching. However, teachers wishing to implement data science-based and individualized analytics solutions, the requirements, benefits, and risks often remain vague. For example, what should the machine learn from its human counterpart, how reliable are the grades or feedback provided by an automated essay evaluation system both in the eyes of a teacher and a student, or what are the consequences of misleading a student with incorrect scores or inappropriate live feedback? These issues depend mainly on the capacity of technology to mimic the human know-how. This research paper explores the application of the most recent technologies in natural language processing and deep learning and provides insight as to whether they improve the accuracy of machine scoring compared to previous research in the area. In addition, this paper aims to determine the number of

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 294–299, 2018.
https://doi.org/10.1007/978-3-319-91464-0_30

sample hand-graded essays necessary per writing construct to obtain reliable levels of performance in automated essay scoring.

2 Research Question and Methodology

Central to this study is one of the eight essay datasets (dataset #8) created in the setting of the 2012 Automated Student Assessment Prize (ASAP) contest hosted by the Kaggle platform and sponsored by the William and Flora Hewlett Foundation. The essays were among the most difficult to predict according to a previous study (Shermis 2014) and had the greatest average number of words. It is important to note that the ASAP datasets are widely used to benchmark the performance of state-of-the-art AES systems (Kumar et al. 2017; Shermis 2014; Zupanc and Bosnić 2017). This study compares the performance of the proposed writing analytics tool against the performance of previously benchmarked AES systems. Other corpora were also available such as the British Academic Writing English (BAWE) corpus, the TOEFL (Test of English as a Foreign Language) corpus, the Cambridge English Corpus, and the International Corpus of Learner English. However, most of them were inadequate when it comes to AES because either (1) they were not free, (2) the scoring grid was not elaborate enough, (3) no grade was provided at all, or (4) contained writings from disparate contexts.

This study explores whether the state of the art in natural language processing and deep learning can improve the capacity of computers in determining the holistic scores of 722 Grade 10 persuasive/narrative/descriptive essays. The essays were written in the setting of a state assessment in the United States. Each essay was graded by two human professional graders using a grid of four rubrics (ideas and content, organization, sentence fluency, conventions). The final score, also called the holistic score, was determined by weighing and summing the rubric scores of the two graders. Special adjudication rules were applicable in certain cases (Kumar et al. 2017). Hence, holistic scores ranged from 10 to 60, inclusively. Figure 1 shows the distribution of essays per score. As it can be noticed, the distribution is biased toward low-quality essays since approximately 73% of the essays have a holistic score between 30 and 40 (40% and 60%). Moreover, the essays have an average number of words of 622.13 words and a standard deviation of 197.08 words.

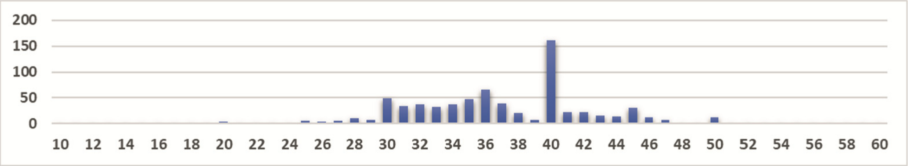


Fig. 1. Essay score distribution with a strong majority (530/722 essays) ranging 30–40 (40–60%).

Each essay was then parsed by the Suite of Automatic Linguistic Analysis Tools (SALAT) (Kyle 2016a) to extract a set of 1,463 baseline writing features[1]. SALAT's

[1] Dataset and code are available at: https://1drv.ms/u/s!ApcQo2VlCqiPmXGh2_zjgDLnPzKp

tools used in this analysis include TAACO, TAALES, TAASSC, and SEANCE (Crossley et al. 2016, 2017; Kyle 2016b; Kyle and Crossley 2015). Essentially, the tools extract and measure (a) 150 indices of local and global cohesion (TAACO), (b) 400 classic and new indices of lexical sophistication (TAALES), (c) many indices related to syntactic development, including classic indices of syntactic complexity and fine-grained indices of phrasal and clausal complexity (TAASSC), and (d) 254 core and 20 component indices based on recent advances in sentiment analysis (SEANCE).

3 Analysis and Results

The first step in this analysis was to normalize the quantitative values of the 1,463 writing metrics (Ng 2017; Rosebrock 2017). The metric distributions were transformed so that the means are 0 and their variances are 1. The goal is to facilitate the convergence of the gradient descent toward optimal local minima in the parameter landscape. The samples were then shuffled and assigned to an 85%-training and 15%-testing dataset. As for the validation set that would drive the tuning of hyperparameters, 5-fold cross-validation was opted for to maximize the number of samples in the training set (Guestrin and Fox 2017) and to balance more fairly the non-uniform distribution of essay scores among the folds.

Next, after having noticed that the trained models suffered from a severe lack of generalizability, the number of writing features was reduced to 128, 96, and 64, those features having the strongest correlations with the essay scores, which resulted in Spearman rank correlation coefficients ranging between 0.30 and 0.50. As for the topology of the neural networks, it varied from 3 to 12 hidden layers and from 64 to 1,024 neurons for the widest hidden layer. At the end, the best neural network architecture consisted of one input layer (96 nodes for the 96 most grade-predictive writing features extracted by SALAT), seven hidden layers (96-64-32-32-16-16-8 for a total of 264 nodes), and one output layer, which predicted the holistic score of an essay (10–60). The loss function underlying the optimization problem was the mean squared error function. Various optimizers were tested including Adam, SGD (Stochastic Gradient Descent), RMSprop, and Adagrad. Learning rates tested included 0.001, 0.005, 0.01, and 0.1, while learning rate decay was set to 0.00025 when applicable. Parameter weights were initialized using techniques such as Xavier/Glorot, He, and normal randomization. Both ReLU (rectified linear unit) and ELU (exponential linear unit) activation functions were tried. Finally, the models were trained during 200–500 epochs, while gradient descent was tested using batch sizes of 4, 8, 16, 32, and 64.

Interestingly, deep learning allowed to train perfect non-linear regressors on the training set. Although the mean squared error was the loss function that drove gradient descent, the performance of models was measured using four other metrics: the quadratic weighted kappa, the percentage of exact matches, the percentage of ± 1 adjacent matches, and the percentage of ± 2 adjacent matches (Kumar et al. 2017). Thus, it was possible to overfit models to the training dataset and reach a quadratic weighted κ value of 1.0 and 100% exact matches and adjacent matches. Obviously, this overfitting was accompanied by a blatant lack of generalization, which was seen by quadratic weighted κ's ranging between 0.0 and 0.2. This meant that the trained models were close to assigning random grades between 10 and 60 to essays it did not learn from. To counter this situation, regularization techniques were leveraged to increase the training error and reduce the validation error. A dropout layer was inserted between the first and second hidden layers with a ratio of 0.5, and L2 regularization with a value of 0.01 was applied to every hidden layer except the first hidden layer.

Table 1 displays the results of 5-fold cross-validation showing the mean squared error, the quadratic weighted kappa, and percentages of exact and adjacent matches for every fold. The quadratic weighted κ ranges between 0.46 and 0.74, the percentages of exact matches vary from 5.7% to 12.3%, the percentages of ±1 adjacent matches are located between 26.2% and 31.1%, and the percentages of ±2 adjacent matches lie in the 41.5%–47.5% range. Table 1 also displays the same metrics for the overall training, validation, and testing sets. The validation entry averages all the values of every metric for every cross-validation fold. It is interesting to note the importance of measuring the performance of predictive models using more than one metric. For instance, Fold 3 has equal or even better performance than Fold 4 for the exact and adjacent match metrics, while having a much lower κ, that is, 0.46 versus 0.65. It shows that inaccurate predictions are much more off the mark in Fold 3 than in Fold 4. Again, from Table 1, it can be seen than the quadratic weighted kappa value significantly decreases from training to validation passing from 0.84 to 0.63 and from validation to testing (from 0.63 to 0.58), while maintaining approximately the same percentages of exact and ±1 adjacent matches, that is, 6.7% to 8.8% for exact matches and 26.6% to 29.5% for ±1 adjacent matches. On the other side, the percentages of ±2 adjacent matches also decrease significantly (53.0% to 38.5%) suggesting and confirming that inaccurate predictions are farther away from actual essay scores for lower quadratic weighted κ's. As an additional resort to improve the predictive performance of the writing analytics tool, an ensemble (bagging) was created out of the five models trained during cross-validation, tuned with a modified dropout rate of 0.25. The ensemble generated a κ of 0.80, 13.8% of exact matches, 37.6% of ±1 adjacent matches, and 57.8% of ±2 adjacent matches.

Table 1. Performance on the training, validation (including cross-validation), and testing sets.

K-fold	MSE	κ	Ex. mat. (%)	±1 adj. mat. (%)	±2 adj. mat. (%)
1	19.4	0.649	10.6%	30.1%	41.5%
2	21.3	0.637	8.1%	28.5%	43.1%
3	38.7	0.455	7.3%	26.2%	41.8%
4	21.7	0.649	5.7%	26.2%	41.8%
5	16.8	0.737	12.3%	31.1%	47.5%
Training	10.3	0.843	6.7%	29.5%	53.0%
Validation	23.6	0.625	8.8%	27.7%	42.9%
Testing	24.1	0.582	7.3%	26.6%	38.5%

Figure 2 draws the training and validation error curves, showing that there is a risk to overfit the neural network model. The validation error curve reaches a plateau at the mean squared error around 25, while the training error curve continues to fall even after 200 epochs of training. Figure 3 displays the learning curves of various training set sizes

(73, 235, 398, 560, 722) obtained by using the same hyperparameters of the best model and 10-fold cross-validation. It confirms that the training and validation error curves reach mean squared errors around 10 and 25, respectively, when training with all the 722 essays. It also seems possible to get a slight improvement, although limited, by increasing the training sample size.

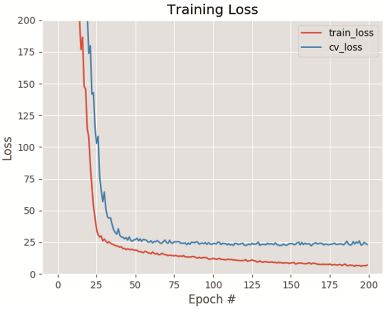


Fig. 2. Training and validation error curves.

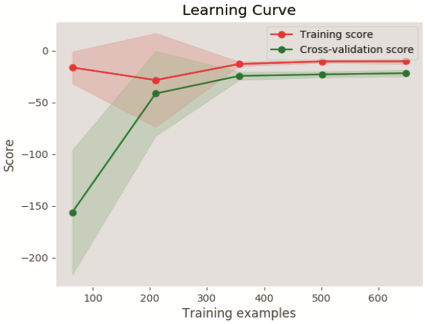


Fig. 3. Learning curve vs. training set size.

4 Discussion and Conclusion

The ASAP essay datasets hosted by Kaggle have often been used in the past as a benchmark dataset to measure the performance of automated essay scoring software. Recently, Zupanc and Bosnić (2017) have developed an automated essay evaluation tool with semantic analysis (a novelty), which included 72 linguistic and content metrics, 29 coherence metrics, and 3 semantic consistency metrics. They tested various regression models such as linear regression, regression tree, neural network, random forest, and extremely randomized tree (ERT). They reported a quadratic weighted kappa value of 0.72 for the neural network for the essay dataset #8, while ERT generated the highest κ, that is, 0.78. It is not clear, however, whether the reported κ values originated from the training, validation, or testing set although they reported having performed a 10-fold cross-validation. The process and results of this cross-validation do not seem to appear anywhere in the paper. Furthermore, the ASAP competition was held in 2012 and since then the validation and testing datasets are no longer available for benchmarking, leaving a dataset of only 722 essays. Originally, the dataset consisted of 1,527 essays, of which 304 essays were used for the testing set and 305 essays were used for the validation set. This certainly impacted the results of the current study as well as the performance reported by Zupanc and Bosnić (2017). In contrast, this study set aside a testing set from the 722-essay dataset and performed 5-fold cross-validation on the training set. This paper reported benchmarks for all three datasets. As indicated in the analysis section, from a mere training perspective, this paper’s approach could perfectly predict the holistic scores of the 722 Grade 10 essays. From a validation and testing perspective, the lower performance could be attributed to the small sample size and to the distribution of essay scores biased toward low-quality essays. Only 3 out of 722 essays had scores greater than 80% (51 out of 60 or higher).

Shermis (2014) reports the results of commercial vendors in the setting of the ASAP competition. Shermis shows that the level of agreement (quadratic weighted kappa) among human raters and the resolved scores (actual essay scores derived by reconciling the disagreement among multiple raters) for dataset #8 is between 0.74 and 0.75. The mean κ for all commercial vendors is 0.67, ranging from 0.60 to 0.73. Using an ensemble technique (bagging), the deep neural network proposed in this paper exhibits a level of agreement (quadratic weighted κ) with essay scores of 0.80. Again, it is important to note that all participants in the ASAP contest had access to a greater sample size than was possible for this study.

The objective of this paper was to provide insight as to the performance that can be expected by applying deep learning in automated essay scoring systems. In addition, it aimed at enlightening the teacher community as to the minimal numbers of hand-graded essays (estimated at between 1,000 and 2,000 essays depending on the score distribution) required per writing construct to craft successful predictive student writing models that could provide real-time tutoring to students. This work also questions whether AES can be deployed in small-scale settings such as a classroom or a school because of the significant workload that would be imposed per teacher, suggesting that such writing analytics tools might be better suited for school boards or large-scale curricula.

References

Crossley, S.A., Kyle, K., McNamara, D.S.: The tool for the automatic analysis of text cohesion (TAACO): automatic assessment of local, global, and text cohesion. Behav. Res. Methods **48**(4), 1227–1237 (2016)

Crossley, S.A., Kyle, K., McNamara, D.S.: Sentiment analysis and social cognition engine (SEANCE): an automatic tool for sentiment, social cognition, and social order analysis. Behav. Res. Methods **49**(3), 803–821 (2017)

Guestrin, C., Fox, E.: Machine Learning: Regression. Coursera (2017). https://www.coursera.org/learn/ml-regression. Accessed 22 Mar 2018

Kumar, V., Fraser, S.N., Boulanger, D.: Discovering the predictive power of five baseline writing competences. J. Writ. Anal. **1**(1), 176–226 (2017)

Kyle, K., Crossley, S.A.: Automatically assessing lexical sophistication: indices, tools, findings, and application. TESOL Q. **49**(4), 757–786 (2015)

Kyle, K.: Suite of Automatic Linguistic Analysis Tools (SALAT) (2016a). http://www.kristopherkyle.com/. Accessed 25 Apr 2018

Kyle, K.: Measuring syntactic development in L2 writing: fine grained indices of syntactic complexity and usage-based indices of syntactic sophistication. Doctoral Dissertation (2016b). http://scholarworks.gsu.edu/alesl_diss/35

Ng, A.: Improving Deep Neural Networks: Hyperparameter tuning, Regularization and Optimization. Coursera (2017). https://www.coursera.org/learn/deep-neural-network. Accessed 22 Mar 2018

Rosebrock, A.: Deep Learning for Computer Vision with Python, 1st edn. PyImageSearch (2017). https://www.pyimagesearch.com/deep-learning-computer-vision-python-book/. Accessed 22 Mar 2018

Shermis, M.D.: State-of-the-art automated essay scoring: competition, results, and future directions from a United States demonstration. Assess. Writ **20**(1), 53–76 (2014)

Zupanc, K., Bosnić, Z.: Automated essay evaluation with semantic analysis. Knowl.-Based Syst. **120**, 118–132 (2017)

A Hybrid Architecture for Non-technical Skills Diagnosis

Yannick Bourrier[1,2(✉)], Francis Jambon[2], Catherine Garbay[2], and Vanda Luengo[1]

[1] Sorbonne Université, CNRS, LIP6, 75005 Paris, France
vanda.luengo@lip6.fr

[2] Univ. Grenoble Alpes, CNRS, Grenoble INP, LIG, 38000 Grenoble, France
yannick.bourrier@lip6.fr,
{francis.jambon,catherine.garbay}@imag.fr

Abstract. Our Virtual Learning Environment aims at improving the abilities of experienced technicians to handle critical situations through appropriate use of non-technical skills (NTS), a high-stake matter in many domains as bad mobilization of these skills is the cause of many accidents. To do so, our environment dynamically generates critical situations designed to target these NTS. As the situations need to be adapted to the learner's skill level, we designed a hybrid architecture able to diagnose NTS. This architecture combines symbolic knowledge about situations, a neural network to drive the learner's performance evaluation process, and a Bayesian network to model the causality links between situation knowledge and performance to reach NTS diagnosis. A proof of concept is presented in a driving critical situation.

Keywords: Neural networks · Bayesian networks · Ill-defined domains

1 Introduction

The goal of our work is to design an Intelligent Tutoring System for learning of NTS in front of critical situations. Learning takes place within virtual environments involving advanced sensing and simulation devices. Two application fields are considered, namely driving and midwifery. NTS are metacognitive abilities that complement technical skills and contribute to safe and efficient task performance [1]. For the learner confronted to critical situations, the challenge is (i) to maintain his/her technical skills and (ii) to maintain the situation's criticality within acceptable bounds, as failure to mobilize NTS is the cause of many accidents [1]. Our team built an ITS that combines knowledge in psychology and computer science to teach NTS to non-novice learners, by making them face a variety of dynamically generated critical situations. We focus in this paper on two main challenges, related to the ill-defined task of learner NTS diagnosis [2, 3]. Firstly, to ensure the soundness of situations, for both learner and machine, by driving the simulator towards situations that lead to accurate experience for the learner and accurate set-up for the machine to learn. Secondly, to cross the semantic gap, i.e. drive the building of the learner's model, from low-level time-stamped indicators to high level diagnosis of NTS. Considering this semantic gap, a hybrid architecture [4] is proposed, combining a

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 300–305, 2018.
https://doi.org/10.1007/978-3-319-91464-0_31

neural network (NN) to drive early diagnosis toward performance evaluation, and a Bayesian network (BN) to drive performance evaluation toward NTS evaluation (explain the variation of performance in terms of NTS deployed to cope with criticality). Section 2 first provides an overview of the domain and the main design choices, then describes the proposed architecture. Section 3 presents a use case for a critical learning situation in driving. We conclude by a recapitulation of the paper's contribution and by suggesting how this model can be used for generation of new, adapted learning situations.

2 Architecture

In both midwifery and driving, diagnosis of NTS is an ill-defined task [3], which has the following characteristics. (1) There are indefinite starting and ending points of a learning situation and overlapping subproblems, since the learner's activity inside the ILE mediates the evolution of the situation. For example, in driving, the initial speed of the driver may change the starting point of a critical situation, and whether a driver decides to overtake an obstacle may expose him to a new danger, such as a car coming from the opposite direction. (2) Technical and non-technical skills overlap. Perceptions and actions are markers of the learner's technical skills being applied in response to a situation, but also of his or her underlying cognitive processes. (3) There are no domain-specific markers of NTS. These markers do exist, but they must be identified by experts beforehand [1]. The architecture we built is designed to handle these ill-defined characteristics. It is a hybrid architecture constituted of three modules, each using a different approach given the nature of the problem it is facing.

2.1 Selecting Time Windows for Performance Analysis

This module allows to target learning situations that we are interested to observe. It uses domain knowledge to select the time windows (called learning situation) where diagnosis will be performed. Two kinds of semantic information are used. First, triggers, are objects of the world (from the virtual reality simulator) whose state is expected to provoke a stereotyped technical response by the learner (e.g.: a red light, or a pedestrian crossing the street). The trigger targets a technical response during a period which we called a phase, the basic time unit where performance is evaluated. Evaluation being framed on these small time periods, it avoids the appearance of long term effects which would complexify it. As such, triggers provide a solution to overlapping subproblems during the learning session. To identify the more "knowledgeable" phases where this evaluation can provide relevant information about the learner's NTS, we introduced a second semantic criterion: precursors of critical situations. These precursors are events provided by the scenario generation module, designed to target the learner's NTS by raising the situation's criticality, without changing which technical response is appropriate. A broken sensor in the birth room, or a father disturbing the midwife's work are good examples of precursors. Contexts where a precursor can be generated define the boundaries of a learning situation. The triggers inside this learning situation define the phases where diagnosis of NTS is performed.

2.2 Analysis of Performance Variations

Studies have shown that increased or decreased worker technical performance during a critical situation, in comparison to the same worker's usual performance during non-critical situations, can be explained by the good or the bad mobilization of the relevant NTS for this situation [1]. This module analyses the variations in a learner's technical performance during normal situations and critical situations, since these variations can be explained by NTS. Technical performance is analyzed using a NN trained using supervised regression, through experts' rankings of learner performance during a phase. This process is replicated for each phase, and separate rankings of the learner's perceptions and actions are obtained. More information about these rankings can be found in [3]. To provide the baseline for performance comparison, a learner's technical performance scores are initially obtained through a bootstrapping period at the beginning of the learning session (that is composed of several learning situations). The learner is confronted to situations during which the precursors do not appear, i.e. the situations are not critical. His perceptions and gestures are analyzed phase after phase. The resulting rankings are averaged to provide the baseline value. Once enough information is acquired to obtain the baseline, precursors can now be generated, so the learner starts facing critical situations. The same process is done during these new situations. The difference between the baseline value and the performance value obtained during a critical situation is computed for both perceptions and actions. Two continuous values are obtained, providing information about the learner's fluctuations in perceptions and actions when facing criticality. These values are integrated in module 3 as evidence nodes to reach NTS diagnosis Fig. 1.

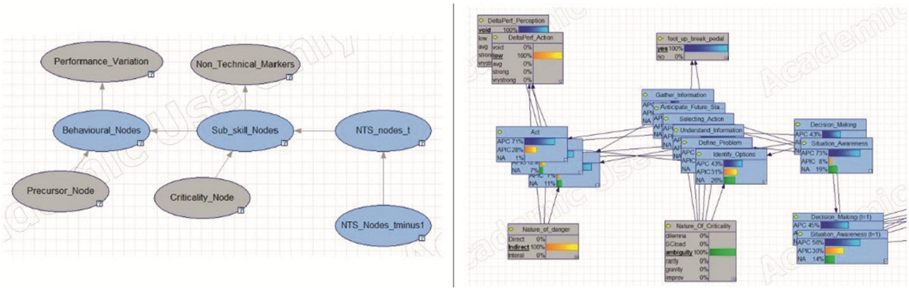


Fig. 1. Meta-model and example of the BN for NTS diagnosis. Evidence nodes are represented in grey, skill nodes in blue. The same inference process takes place for each temporal NTS node.

2.3 Bayesian Network for NTS Diagnosis

This module uses knowledge about the learning situations to explain a learner's variations in performance by several underlying cognitive processes, and reach NTS diagnosis, through a dynamic BN, which is updated phase after phase. It accumulates evidence about the learner's variations in perception and action performance, to diagnose the learner's NTS and their temporal evolution. We used two kinds of important information which are modelled in our BN. (1) Evidence about the nature of a critical situation, as identified by researchers in psychology and ergonomic [5], and (2) Flin et al's taxonomy of NTS cognitive sub processes [1].

The BN uses two different kinds of **evidence nodes**. Firstly, *Learner Nodes* provide information about the learner's actions. They are of two kinds. Performance variation nodes are obtained through module 2. Non-technical markers (NTM) provide additional evidence in the form of a Boolean value, about the good or bad mobilization of certain subskills, when the learner faces well known critical situations where experts were able to perform CTA. Secondly, *Situation Nodes* provide information about the phase in which the learner's behavior is taking place. They filter the cognitive processes which are relevant for diagnosis in that phase. Situation knowledge is modeled as a two-dimensional value, characterized by the nature of its criticality, and the nature of its precursor. Each generated critical situation is defined by this couple and is designed to target specific subskills. The precursor node provides information about the kind of precursor which is currently being used. A precursor can be of three kinds: direct (can be seen by the learner), indirect (hidden, must be deduced from the context), or internal (triggered by the learner's actions). The criticality node provides information about the nature of the criticality which is faced. Researchers identified six different criticality dimensions which can be generated by the scenario generation module: dilemma, socio-cognitive load, ambiguity, rarity, gravity, and impermissibility [5]. **Skill nodes** allow the BN to model the causal dependencies separating learner performance and NTS. Flin et al's taxonomy [1] is widely used by domain experts in the medical domain, to describe lower level cognitive processes, which are easier to identify than high level skills when performing CTA. For example, a skill such as situation awareness can be decomposed into three sub processes, which are: (i) gathering information, (ii) recognizing and understanding information, and (iii) anticipating future states [1]. Relations between these sub-skills and NTS are easy to represent inside a BN. However, it is more complex to cross the gaps between these subskills and the evidence nodes about the variations in perception and action performance. Therefore, we included an intermediate set of nodes which we call behavioral nodes: perceive, understand, and act. These nodes improve the hierarchical structure of the BN and clarify the relations between sub-skills and variations in performance. Moreover, they are an intermediate step whose high semantic significance allow to represent links between different NTS subskills. Skill nodes can take three values: applied correctly (APC), applied incorrectly (APIC), or not relevant in this phase (NA). Information determining the probability tables for each skill node comes from evidence nodes.

3 Proof of Concept for an Ambiguous Driving Situation

Figure 2 represents a critical learning situation in driving. The association of a truck and a crosswalk is a precursor, raising the situation's ambiguity. Module 1 separates this critical learning situation in 2 phases, given the situation's triggers. Phase 1 starts when the first trigger (the truck) is on sight. Phase 2 begins when the second trigger (the pedestrian) is not dissimulated by the truck anymore and ends once the pedestrian has finished crossing the street. Module 2 analyses performance at the end of each phase, given the learner's traces inside the VRE and from the eye-tracking device. The two values are compared to the learner's performance baseline. The resulting values are provided as evidence nodes to the BN. Table 1 presents the results provided by the BN for the first phase of the situation presented in Fig. 2, given different degrees of learner performance. For this situation, the criticality node was filled at "ambiguity" and the precursor node at "indirect" as per identified by experts for this set-up.

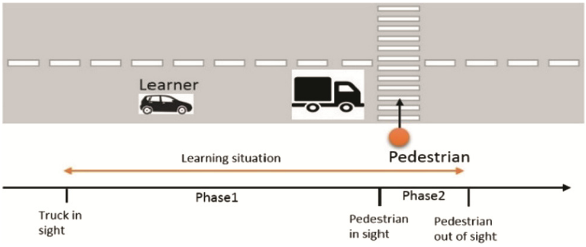


Fig. 2. Representation of the learning situation and the two phases based on the trigger states.

Table 1. Examples of values obtained for phase 1 of the situation presented in Fig. 2. Three degrees of performance variations are tested. Learner1 reacted accordingly to the critical situation, learner2's performance was notably worse than his baseline results, and learner3's actions and perceptions were strongly perturbated. For each learner, three cases are proposed. The 1st case shows the resulting diagnosis when no NTM are used. The 2nd case considers one NTM correctly applied by the learner. The 3rd case considers the same NTM incorrectly applied.

learner ID	learner1	learner1	learner1	learner2	learner2	learner2	learner3	learner3	learner3
delta perception	very low	very low	very low	avg	avg	avg	very high	very high	very high
delta action	low	low	low	low	low	low	high	high	high
foot up brake pedal (NTM)	NA	APC	APIC	NA	APC	APIC	NA	APC	APIC
Situation Awareness	P(APC) = 0.58 P(APIC) = 0.17 P(NA) = 0.26	P(APC) = 0.73 P(APIC) = 0.08 P(NA) = 0.19	P(APC) = 0.30 P(APIC) = 0.33 P(NA) = 0.37	P(APC) = 0.38 P(APIC) = 0.28 P(NA) = 0.34	P(APC) = 0.16 P(APIC) = 0.45 P(NA) = 0.39	P(APC) = 0.58 P(APIC) = 0.13 P(NA) = 0.29	P(APC) = 0.15 P(APIC) = 0.50 P(NA) = 0.35	P(APC) = 0.05 P(APIC) = 0.64 P(NA) = 0.31	P(APC) = 0.40 P(APIC) = 0.17 P(NA) = 0.43
Decision Making	P(APC) = 0.43 P(APIC) = 0.25 P(NA) = 0.31	P(APC) = 0.42 P(APIC) = 0.26 P(NA) = 0.31	P(APC) = 0.43 P(APIC) = 0.24 P(NA) = 0.32	P(APC) = 0.40 P(APIC) = 0.27 P(NA) = 0.33	P(APC) = 0.41 P(APIC) = 0.26 P(NA) = 0.33	P(APC) = 0.39 P(APIC) = 0.28 P(NA) = 0.33	P(APC) = 0.25 P(APIC) = 0.41 P(NA) = 0.34	P(APC) = 0.25 P(APIC) = 0.41 P(NA) = 0.34	P(APC) = 0.25 P(APIC) = 0.41 P(NA) = 0.34

The BN provides an estimation of a learner's NTS given his performance during various situations having different characteristics. In the example of Table 1, diagnosis strongly variates given different performances and given the presence or absence of a NTM, which suggests that the model is already quite sensible even though multinomial nodes were used as input. Moreover, the presence of a NTM strongly influences the target probabilities. This is to be expected as NTMs are related to sub-skills which makes the information propagate more easily towards NTS nodes. Finally, we can observe that most of the variations take place for situation awareness, and not decision making, because the information provided by the situation nodes (i.e.: nature of precursor being "indirect" and nature of criticality "ambiguous") were described by experts as more influential towards situation awareness sub-skills than decision making.

4 Conclusion

We presented an architecture able to diagnose NTS from a learner's perceptions and actions, in technical domains where critical situations occur. We first described the ill-defined aspects of this design task, and then presented a hybrid approach of three modules designed to handle these problems. The main limitation of this architecture is its focus on situation awareness and decision making, as these NTS have a similar importance in the two domains we applied our architecture to. Social skills, at the opposite, have very different degrees of importance between medicine and driving. Moreover, they may require different tools to be correctly analyzed, such as speech recognition modules for example. A first next step is to replace the currently filed-by-expert conditional probability tables of the BN by more accurate parameter learning. The biggest next step is to make use of the diagnosis process described in this paper for our system to make decisions about the kinds of feedback which should be given to the learner, and the generation of the next learning situation. We currently aim at exploring the association of a short-loop feedback system, able to provide real-time feedback during the experience of a learning situation, and a long-loop feedback able to consider the learner model and knowledge about the situations to improve post-situation feedback, and to generate new critical learning situations.

Funding. This research is funded by the French National Research Agency (ANR) via the MacCoyCritical Project (ANR-14-CE24-0021).

References

1. Flin, R.H., O'Connor, P., Crichton, M.: Safety at the sharp end: a guide to non-technical skills. Ashgate Publishing Ltd., Farnham (2008)
2. Lynch, C., Ashley, K., Aleven, V., Pinkwart, N.: Defining Ill-defined domains; a literature survey. In: Proceedings of the Intelligent Tutoring Systems for Ill-Defined Domains Workshop, ITS 2006, pp. 1–10 (2006)
3. Bourrier, Y., Jambon, F., Garbay, C., Luengo, V.: An approach for the analysis of perceptual and gestural performance during critical situations. In: Lavoué, É., Drachsler, H., Verbert, K., Broisin, J., Pérez-Sanagustín, M. (eds.) EC-TEL 2017. LNCS, vol. 10474, pp. 373–378. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66610-5_29
4. Fournier-Viger, P., Nkambou, R., Nguifo, E.M.: Building intelligent tutoring systems for ill-defined domains. In: Nkambou, R., Bourdeau, J., Mizoguchi, R. (eds.) AITS. Studies in Computational Intelligence, vol. 308, pp. 81–101. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-14363-2_5
5. Burkhardt, J.M., Corneloup, V., Garbay, C., Bourrier, Y., Jambon, F., Luengo, V., Job, A., Cabon, P., Benabbou, A., Lourdeaux, D.: Simulation and virtual reality-based learning of non-technical skills in driving: critical situations, diagnosis and adaptation. IFAC-PapersOnLine **49**(32), 66–71 (2016)

A Novel Learning Early-Warning Model Based on Random Forest Algorithm

Xiaoxiao Cheng, Zhengzhou Zhu(✉), Xiao Liu, Xiaofang Yuan,
Jiayu Guo, Qun Guo, Deqi Li, and Ruofei Zhu

Peking University, Beijing, China
zhuzz@pku.edu.cn

Abstract. The learning early-warning is an effective way to optimize the teaching effect and teach students in accordance of their aptitude. At present, the learning early-warning faces low accuracy, high value of MSE and MAE. We propose a novel learning early-warning model: LEWM-RFA. The model divides students' learning behaviors data into three dimensions: knowledge, behavior and attitude. Then the model uses random forest algorithm to extract features that can affect students' grades, and then predicts students' final exam scores. Students are divided into three warning levels according to their grades. Compared with the model based on the linear regression algorithm, the LEWM-RFA's MSE decreases by 27.498% and the LEWM-RFA's MAE decreases by 26.960%.

Keywords: Learning early-warning · Random forest algorithm
Prediction

1 Introduction

Learning early-warning model collects and analyzes the data of learning process to accurately determine the student's learning situation and status. It can timely detect learning problems and warn [1]. It is a set of mechanisms, including procedural prediction, evaluation and processing [2]. This method can solve the problem of shortage of teachers and help teachers to accurately grasp the student's situations and learning status. Then, it can provide students with early-warning and targeted intervention. Thus, we can achieve the effect of individualized teaching. In the process of learning, teachers can fully grasp the students' learning situation and we cannot ignore the effect of targeted early-warning and intervention on the students. The random forest is an integrated classifier composed of a set of decision tree classifiers. Under given independent variables, each decision tree classifier determines the best classification result by voting [3]. The random forest algorithm can effectively extract the highly influential features from the structured data, and can quickly obtain the influence weight of each feature. There are two contributions of this paper: (1) we design an off-line and on-line data collection method to fully obtain the students' knowledge, behaviors and attitudes; (2) we design a learning early-warning model based on the random forest algorithm (LEWM-RFA). The validity of the model is verified by the application in actual environment. Compared with the learning early-warning model based on linear

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 306–312, 2018.
https://doi.org/10.1007/978-3-319-91464-0_32

regression algorithm (LEWM-LRA), the mean square error reduces by 27.498% and the average absolute difference reduces by 26.960%.

2 Related Works

There are two approaches of studying early-warning model, namely post-warning and pre-warning. Post-warning means that alert would be triggered if the value is more than or less than the preset threshold. It can be divided by two ways, and the first type is to set thresholds respectively for each course grades, times of absenteeism, financial situation and school attendance each day [2, 4, 7]. Considering students' data of daily study, course selection, attendance, performance and psychological status comprehensively, students can be divided into three ranges, corresponding to three different warning signals [5, 6]. Pre-warning refers to predicting students' learning behavior performance or academic success, identifying students at risk in advance based on the student's historical data. Various approaches have been trialed by scholars on the study of pre-warning. The first way is the Course Signals Project developed at Purdue. The main idea of SSA gives the weights to curriculum performance, curriculum effort, pre-academic achievement and the student's characteristics and obtain the final forecast score. Secondly, the main online behaviors extracted from students' behavior data to establish a relevant index structure. The combination of fuzzy theory and the Analytic Hierarchy Process (AHP) was used to build a predictive model to sure the student's performance and find out the "difficult" student in advance [8].

3 Learning Early-Warning Model

The learning early-warning is based on the data mining of knowledge, behavior and attitudes generated by students' learning process. It uses learning and analysis techniques to understand students' learning situation and assess students' learning status. Learning early-warning aims to identify the "learning crisis" which predicts students may fail in the final exam as early as possible, find out the students who have learning crisis to alert themselves and improve the passing rate of the course. In this paper, the aim of learning early-warning model is to improve the students' scores and passing rate of courses. Based on the goal, we design the model as follows.

- **Extract early-warning features**
  In the random forest classification prediction, an important task is to find relevant important features. By comparing the importance of features generated by random forest, all features are ranked. We use the sequence backward search method when searching for a subset of features that achieves the maximum classification accuracy rate. Sequencing backward search method refers to sorting features using the variable importance metric of random forest algorithm, removing one least significant feature from the feature set, iterating one by one, and calculating the classification accuracy, finally getting the least number of variables, the highest classification accuracy as the feature set. The specific feature extraction process is as follows.

```
Input: raw data set Data0
Output: maximum classification accuracy AccruacyMax and the corresponding characteristic variable RFSort

Step 1: Original data set is divided into the training set Train0 and the test set Test0;
Step 2: Set the initial classification accuracy Accruacy0 = 0, the maximum classification accuracy AccruacyMax;
Step 3: Run the RFA on Train (i) to build the algorithm model RFModel (i);
Step 4: Use RFModel (i) to perform the classification process on the test set Testi;
Step 5: Calculate the accuracy of this round of classification Accuracy (i);
    if (Accuracy (i) > AccuracyMax)
    AccuacyMax = Accuracy (i);
Step 6: Sort the feature variables by importance to get RFSort (i), remove the least important variable;
Step 7: Get new data set Data (i + 1) and re-divide it into Train (i + 1) and Test (i + 1);
Step 8: Perform Step 3.
```

- **Construct a learning early-warning model**

(1) **Divide the data set.** In this paper, the students' learning data and test scores extracted from the feature are divided into two data sets according to the student ID, 70% for the training set and 30% for the test set. The students' learning data and course test scores are used to train LEWM-RFA. The test set is used to compare with the results generated by model and evaluate the model.

(2) **Train the model. (a)** Training set is sampled by the bootstrap method to form a new training set. **(b)** Use the CART algorithm to set a complete decision tree without pruning. **(c)** Repeat (a) and (b) until establish K decision trees. The random forest structure is completed.

(3) **Forecast score. (a)** Predict the performance of each variable x of the test set. Each one of the k decision trees votes on the performance of variable x. **(b)** Calculate all votes H(x). The average of H(c) is variable x's final forecast. The output is the predicted score for each variable x in the test set.

(4) **Classification of warning levels.** According to the model results, we can obtain the influence weights of how grades of students are affected by the warning factors used in LEWM-RFA. The highest N influence weight variables are important aspects of warning object for teachers to focus on. Mean absolute error (MAE) of random forest learning prediction model prediction results is as shown in formula (1):

$$\mathrm{MAE} = (|\Delta 1| + |\Delta 2| + \ldots + |\Delta \mathrm{n}|)/\mathrm{n} \tag{1}$$

In (1), △ is mean absolute error; △1, △2…, △n are respectively the absolute error measured by the model. The passing score of the course is Passing-Score, abbreviated as PS, and the full course is Full-Score, abbreviated as FS. There are some errors in the prediction results of early warning model. [(PS − MAE), (PS + MAE)] is the range of forecast scores where real scores are near to the pass line. [0, (PS − MAE)] is the range of forecast scores where the failing rate of real scores is extremely high. [(PS + MAE), FS] is the range of forecast scores where the failing rate of real scores is extremely low. According to the results, the warning levels are divided into three levels: red, yellow and green, and the rules are shown in Table 1.

Table 1. Classification of warning levels

Classification of warning levels	Red alert level	Yellow alert level	Green alert level
Meaning of each level	There is a great learning crisis, a great probability of failure	There is a learning crisis, there may be failure	Minimal learning crisis, minimal probability of failure
Forecast range of grades	[0, (PS − MAE)]	[(PS − MAE), (PS + MAE)]	[(PS + MAE), FS]
Warning degree	Strongest warning & guidance	Moderate warning & guidance	Weakest warning and guidance

According to the results, the warning levels of the students are computed in the test set, and then the students with different warning levels are given different degrees of warning and targeted guidance based on the previous results.

- **Evaluate the learning early-warning model**

The evaluation indexes of the learning early-warning model can be listed as follows: the accuracy rate of predicted warning degrees naming Accuracy, the Mean Squared Error (MSE) of predicted scores and MAE. The Accuracy is the ratio of the number of correct warning degree predictions to the total number of predicted students, MSE can be provided by the model directly, and MAE is calculated by formula (1). We can also evaluate the LEWM-RFA in the practical application by the increase of passing rate and the feedback of the early-warning object.

4 Application and Verification

- **Application environment**

The application environment is a hybrid learning environment that combines online learning platform with offline class. There are 59 s bachelor degree students of Peking University and the course is Software Engineering which is compulsory for the major SE.

The actual data is the learning data of 7 weeks from April 13 to June 1, 2017. The students' learning activities data recorded by the online learning platform and the data of the offline learning were considered as potential learning early-warning factors.

- **Extraction of early-warning features**

The practical approach uses the variable importance measure of the random forest algorithm to sort the features. The sequence backward search method is used to remove the least important feature (the least importance score) from the feature set, calculate classification accuracy, and get the smallest number of variables which is the highest classification accuracy of the feature set as a feature selection result.

- **The construct of learning early-warning model**

The division of the data set: 59 students have studied this course. 58 students have got the real grades with one absence in the final exam. The 58 students' learning data and the final grade data are divided into two sections, the training set (Train): data of No.1–No.40 students and the testing set (Test): data of No.41–No.57 students.

(1) **The division of the warning levels.** According to the standard of our test:100 is the max score, 60 is the passing score, MAE value is 6.7 and red, yellow, green warning intervals are: Red: [0, 52.3] Yellow: [52.4, 66.7] Green: [66.8, 100]; Listing the warning interval for the predicted and true scores of the testing set, as shown in Table 2.

Table 2. Comparison between real score and predictive score of LEWM-RFA

Student ID	Real score	Predictive score	Student ID	Real score	Predictive score	Student ID	Real score	Predictive score
754	92[Green]	85.3[Green]	764	100[Green]	94.3[Green]	772	85[Green]	85.8[Green]
756	84[Green]	88.0[Green]	765	96[Green]	89.1[Green]	773	91[Green]	91.3[Green]
757	94[Green]	89.6[Green]	766	47[Red]	74.1[Green]	776	84[Green]	89.4[Green]
759	92[Green]	88.6[Green]	769	83[Green]	83.6[Green]	777	88[Green]	85.6[Green]
760	83[Green]	89.6[Green]	770	93[Green]	90.3[Green]	778	63 [Yellow]	86.6[Green]
763	85[Green]	90.4[Green]	771	68[Green]	77.1[Green]	779	82[Green]	88.7[Green]

(2) **Evaluation of learning early-warning model.** Based on the same application data and settings, we use LEWM-LRA to predict the results, and the MAE in LEWM-LRA is 9.173, the red-yellow-green interval is [0, 52.7] [50.827–69.173] [69.173, 100]; The results is shown in Fig. 1.

In our case Accuracy is defined as the ratio of the number of samples correctly sorted by the classifier to the total number of samples for a given test dataset. In this application, we correctly classify 16 samples with a total forecast of 18, so the Accuracy value is 16/18 = 88.89%. Identically the Accuracy value of LEWM-LRA was 16/18 = 88.89%. Indices of LEWM-LRA and LEWM-RFA are showed in Table 3. In the MAE comparison, the MAE of the LEWM-RFA is 6.7, which is

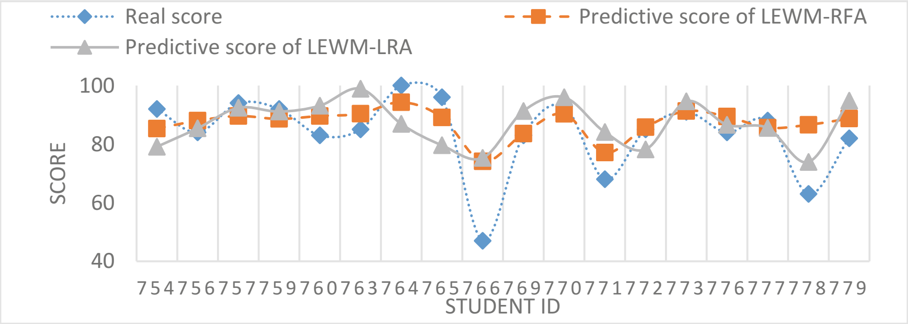


Fig. 1. Real score and predictive score. (Color figure online)

Table 3. The accuracy, MAE and MSE of LEWM-LRA and LEWM-RFA

Index set	LEWM-LRA				LEWM-RFA			
	Total	<50.827	50.827–69.173	69.173–100	Total	<53.3	53.3–66.7	66.0–100
Accuracy	88.89%	0	50%	100%	88.89%	0	50%	100%
MAE	9.173				6.700			
MSE	130.801				94.833			

significantly smaller than EWM-LRA. The MAE value is 9.173, which is 26.96% lower than the other one. As for the MSE value, the MSE value of this model is 94.833, which is obviously smaller than EWM-LRA. The MSE value is 130.801, which is 27.498% lower. Thus, LEWM-RFA has a better performance.

5 Conclusions and Future Work

We present a novel model of learning early-warning: LEWM-RFA. It extracts features used random forest algorithm, predicts students' final exam results, divides students into three early-warning levels according to the prediction, presents different levels of warning to different levels of students, and finally enhances the pass rate of the course. Comparing with the LEWM-LRA, LEWM-RFA's MSE is reduced by 27.498% while the MAE reduced by 26.960%. The effect of model LEWM-RFA is still available.

Acknowledgements. This paper was supported by National Key R&D Program of China (Grant No. 2017YFB1402400), National Natural Science Foundation of China (Grant No. 61402020), and CERNET Innovation Project (Grant No. NGII20170501).

References

1. Wang, L.: Design of online learning early-warning model based on big data-the learning early-warning of “research and practice column about big data in education”. Mod. Educ. Technol. 5–11 (2016)
2. Hua, J.: Learning Early-Warning Model: Experience from Universities in Taiwan. Jiangsu Higher Education, pp. 136–138(2007)
3. Breiman, L.: Random forests. Mach. Learn. **45**, 5–32 (2001)
4. Zhang, H.: Exploration of Learning Early-warning Mechanism for Universities. Science and Technology Information, pp. 811–966(2009)
5. Xu, Q., Zu, Y.: Construction and Effect of Learning Early-warning and Assistance Mechanism for Undergraduate Students in China. Science and Technology Information, pp. 547–591 (2009)
6. Xu, W., Yang, Y.: Construction and design of the learning early-warning system: experience from Red River College. China Education Info, pp. 94–96 (2017)
7. Huaining, S.: Research and design of the students study warning system model based on campus network. Sci. Mosaic 35–37 (2011)
8. Gu, X., Liu, Y., Hu, Y.: Linking learning analytics with instruction practices: approach to the data-enabled research to learning enhancement. Open Educ. Res. **22**, 34–45 (2016)

Improving Inference of Learning Related Emotion by Combining Cognitive and Physical Information

Ernani Gottardo[1(✉)] and Andrey Ricardo Pimentel[2]

[1] Federal Institute of Education, Science and Technology of Rio Grande do Sul - IFRS, Erechim, Brazil
ernani.gottardo@erechim.ifrs.edu.br

[2] Federal University of Paraná - UFPR, Curitiba, Brazil
andrey@inf.ufpr.br

Abstract. Researches in areas such as neuroscience and psychology indicate that emotions directly impact learning. So, adapting to the learners' affective reactions became a requirement and also a challenge for building a new generation of affect aware computing learning environments. In this paper, we present a hybrid approach for inferring learning related emotion that combines cognitive and physical data, gathered using minimal or non intrusive methods. In an initial experiment with students in a real education environment it was possible to obtain promising results when comparing some usual performance metrics with correlated works. In this study we achieved accuracy rates and Cohen's Kappa near to 65% and 0.55, respectively. Furthermore, considering the open and expansible nature of this proposal, we believe that this results could be improved in the future by adding new data or new sensors to the model, for example.

Keywords: Affective Computing · Emotion Inference
Emotion and Learning · Adaptive Systems

1 Introduction

One of the main limitations currently presented by educational software like Intelligent Tutoring System (ITS) is the lack of features to adapt to students' emotional states [1,5]. Much of nowadays ITS pay little or no attention to the emotional experience of students and because of this do not reach the full level of interactivity as humans tutors can do [3]. This limitation becomes relevant considering the inextricable relationship between emotions and learning [12].

Advances in the area known as Affective Computing have motivated a growing body of research in an effort to develop systems that could recognize and adapt to students' affective states [7]. Despite these advances, accuracy of inferred emotion are not yet sufficient to be used as the bases for adaptation, particularly in real classroom learning environments [3]. There is also the practical challenge of deploying some of the physical affective sensors [1,3].

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 313–318, 2018.
https://doi.org/10.1007/978-3-319-91464-0_33

In this context, this paper presents a proposal and implementation of a Hybrid Inference Model of Learning Related Emotions - ModHEmo. This proposal stand out by presenting an approach that combines physical and cognitive data, gathered with minimally or non intrusive sensors that could be easy deployed out-of-the-lab. This approach is grounded on the fact that emotions in humans are strongly related with some physical reactions, but also include a rational and cognitive process [12].

2 Hybrid Emotion Inference Model

As this proposal focus on educational software, we need to deal with emotion that impact and correlate with learning outcomes. Priors works [2–4,8,11] show that there is no consensus about a specific set of learning related emotion. So, to choose the set of emotions to infer we used as conceptual foundations the work of [14]. This work is based on two consolidated theory: the 'circumplex model' [13] and the 'spiral learning model' [9]. These theories define a two dimensional space in which the emotions of a student could dynamically move during learning. Based on these references, we decide to implement the inference process considering not a specific set of emotions, but grouping them into quadrants.

Figure 1 shows the approach used in this work to represent the learning related emotions in a two dimensional space. In this proposal the 'Valence'(horizontal axis) and 'Arousal'(vertical axis) dimensions are used to position the emotions in the "Q1", "Q2", "Q3" and "Q4" quadrants, more a "Neutral" state named "QN". These quadrants played the role of classes in the classification processes performed by the ModHEmo that will be described in the next section.

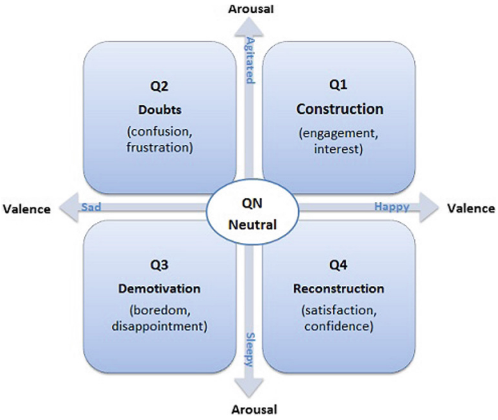


Fig. 1. Quadrants and learning related emotions

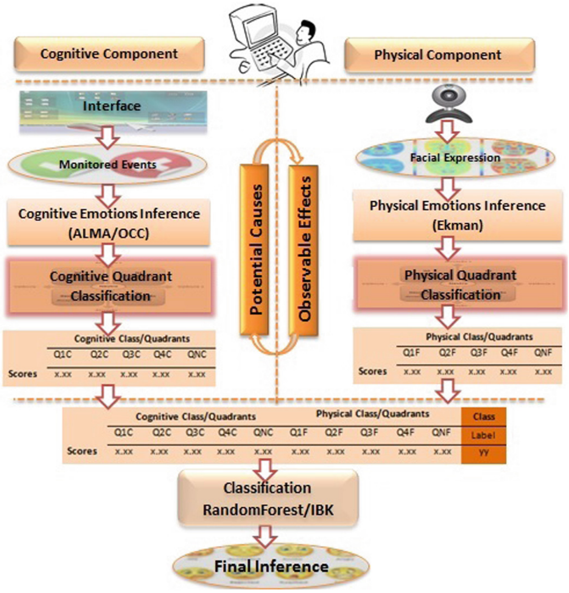

Cognitive Component
Physical Component
Interface
Monitored Events
Cognitive Emotions Inference (ALMA/OCC)
Cognitive Quadrant Classification
Cognitive Class/Quadrants
Q1C Q2C Q3C Q4C QNC
Scores x.xx x.xx x.xx x.xx x.xx
Potential Causes
Observable Effects
Facial Expression
Physical Emotions Inference (Ekman)
Physical Quadrant Classification
Physical Class/Quadrants
Q1F Q2F Q3F Q4F QNF
Scores x.xx x.xx x.xx x.xx x.xx
Cognitive Class/Quadrants
Physical Class/Quadrants
Class
Q1C Q2C Q3C Q4C QNC Q1F Q2F Q3F Q4F QNF Label
Scores x.xx x.xx x.xx x.xx x.xx x.xx x.xx x.xx x.xx x.xx yy
Classification RandomForest/IBK
Final Inference

Fig. 2. ModHEmo's structure and sub-components (labeled with capital letters)

The Fig. 2 schematically presents ModHEmo where the inference process is divided into two main components: physical and cognitive. The final inference of probable students' affective state is made by combining the information of these two components.

The physical component inference is based on the classical Ekman's model [6] and include the eight basic emotions, which are: anger, disgust, fear, happy, sadness, surprise, contempt and neutral. In the cognitive component were considered the eight emotions of the Ortony, Clore and Collins - OCC theory [10] that have impact on the agent itself (e.g. student) and that are triggered as valenced (positive or negative) reaction in response to events. The eight cognitive emotion are: joy, distress, disappointment, relief, hope, fear, satisfaction and fears confirmed.

The cognitive component, based on OCC theory, is responsible for handling relevant events in the computing environment. The physical component deal with observable reactions, using students' face images gathered using a standard Webcam, after the occurrence of a relevant event. These two components returns scores for each emotions of the physical and cognitive components. Further, each one of the components perform a mapping of the emotions' scores for the respective quadrants (see Fig. 1 based on the values of the valence and arousal dimensions.

The final fusion process is accomplished by creating a single dataset containing quadrants scores of each component. After the fusion, this dataset contains 10 attributes (5 physical + 5 cognitive) with scores of quadrants (plus Neutral) for each component. The Class obtained through the labeling process (to be described in the next section) is also part of the dataset. Based on this dataset, we train and test two classification algorithms which perform the inference of the final result of the model.

3 Experiment Design and Results

To check the ModHEmo's performance in a real educational environment, an experiment was conducted with 15 elementary students (ages between 11 and 15 years). In the experiment, the students used a customized version of the educational software 'Tux, of Math Command' or TuxMath[1]. TuxMath is an educational game that allows kids to exercise their mathematical reasoning.

While students used TuxMath, some of the main events of the game were monitored. When a monitored event occurs, an image of the students' face was captured and used as the input to physical component of ModHEmo. The kind of event serves as input for cognitive component.

After completing the game, students labeled the events using a customized tool in order to build a ground truth dataset. This tool allows the student to review the game section, synchronized with a video that shows their facial reactions captured by the webcam. The tool automatically stop the video when some monitored event has occurred and ask students to select a quadrant (represented by emoticons) that best describe their affective state at that moment.

A ground truth with 935 instances of monitored events was created and the classes distribution was 141, 188, 173, 130 and 303 for Q1, Q2, Q3, Q4 and QN, respectively. These dataset was used for training and testing (10-fold cross validation) the classification algorithms RandomForest and IBK using Weka[2]. These algorithms were chosen due their simplicity and performance. The algorithm RandomForest achieve accuracy rate of 64.81% and Cohen's Kappa of 0.545. The IBK accuracy rate was 63.53% and Cohen's Kappa 0.532.

Another useful tool to analyze classifiers performance are ROC (Receiver Operating Characteristic) curves that depict the performance of a classifier without regard to class distribution or error costs. The Fig. 3 depict the ROC curves for the five classes obtained by the RandomForest algorithm (the curves for IBK are very similar). This figure also show the Area Under Curve (AUC) computed in Weka.

In this work, we considered the premise that the combination of the physical and cognitive components could be an effective approach to improve the inference results. Thus, tests were performed to verify the impact in the inference process of using each of the components individually.

To perform this test we created two datasets: one with physical and other with cognitive attributes only. Using only cognitive attributes, the accuracy was respectively 39.25% and 40% for RandomForest and IBK. With only physical attributes the accuracy as respectively 55.29% and 52.19% for RandomForest

[1] http://tux4kids.alioth.debian.org/tuxmath/index.php.

[2] https://www.cs.waikato.ac.nz/ml/weka.

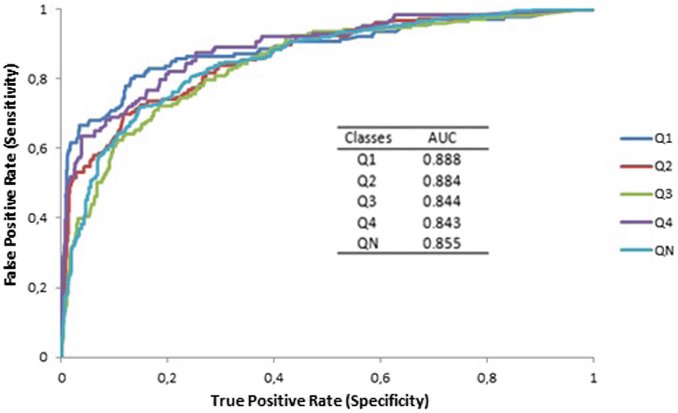


Fig. 3. ROC curves and AUC for RandomForest algorithm

and IBK. This result indicates that the combination of the two ModHEmo's components was important for improving the inference accuracy.

The results achieved in the experiment with ModHEmo presented above show some improvements when compared with [2,3,11]. In our experiment, Cohen's Kappa index was 0.545 and 0.532 for RandomForest and IBK, respectively and AUC value was between 0.843 and 0.888 (see Fig. 3).

4 Final Considerations, Limitations and Future Works

Inferences obtained with this model could be very useful for implementing learning environments or ITS able to appropriately recognize and respond to learners' emotional reactions. The model described in this paper stand out by presenting a method to combine quite distinct information (physical and cognitive) that is little explored in the research community nowadays.

Even considering some limitations, the initial results obtained can be considered promising, since the results obtained are similar or superior to the state of the art. Furthermore, we believe that our hybrid approach resembles the natural process of emotions inference, thus presenting promising opportunities for future improvements by adding new data or sensors.

As future work wed intend to expand the current experiment involving more students with other age groups and also other types of educational environments. It is also intended to evaluate the result of the adding new information in the physical and cognitive components.

Acknowledgments. Authors would like to thank the IFRS and UFPR for financial support of this work.

References

1. Baker, R.S., Gowda, S., Wixon, M., Kalka, J., Wagner, A., Salvi, A., Aleven, V., Kusbit, G., Ocumpaugh, J., Rossi, L.: Sensor-free automated detection of affect in a cognitive tutor for algebra. In: Educational Data Mining 2012 (2012)
2. Bosch, N., Chen, Y., D'Mello, S.: It's written on your face: detecting affective states from facial expressions while learning computer programming. In: Trausan-Matu, S., Boyer, K.E., Crosby, M., Panourgia, K. (eds.) ITS 2014. LNCS, vol. 8474, pp. 39–44. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-07221-0_5
3. Botelho, A.F., Baker, R.S., Heffernan, N.T.: Improving sensor-free affect detection using deep learning. In: André, E., Baker, R., Hu, X., Rodrigo, M.M.T., du Boulay, B. (eds.) AIED 2017. LNCS (LNAI), vol. 10331, pp. 40–51. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-61425-0_4
4. Conati, C.: Combining cognitive appraisal and sensors for affect detection in a framework for modeling user affect. In: Calvo, R.A., D'Mello, S. (eds.) New Perspectives on Affect and Learning Technologies. LSIS, vol. 3, pp. 71–84. Springer, New York (2011). https://doi.org/10.1007/978-1-4419-9625-1_6
5. D'Mello, S., Lehman, B., Sullins, J., Daigle, R., Combs, R., Vogt, K., Perkins, L., Graesser, A.: A time for emoting: when affect-sensitivity is and isn't effective at promoting deep learning. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6094, pp. 245–254. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13388-6_29
6. Ekman, P.: An argument for basic emotions. Cogn. Emot. **6**(3–4), 169–200 (1992)
7. Grafsgaard, J.F., Wiggins, J.B., Boyer, K.E., Wiebe, E.N., Lester, J.C.: Embodied affect in tutorial dialogue: student gesture and posture. In: Lane, H.C., Yacef, K., Mostow, J., Pavlik, P. (eds.) AIED 2013. LNCS (LNAI), vol. 7926, pp. 1–10. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39112-5_1
8. Jaques, N., Conati, C., Harley, J.M., Azevedo, R.: Predicting affect from gaze data during interaction with an intelligent tutoring system. In: Trausan-Matu, S., Boyer, K.E., Crosby, M., Panourgia, K. (eds.) ITS 2014. LNCS, vol. 8474, pp. 29–38. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-07221-0_4
9. Kort, B., Reilly, R., Picard, R.W.: An affective model of interplay between emotions and learning: reengineering educational pedagogy-building a learning companion. In: 2001 Proceedings of the IEEE International Conference on Advanced Learning Technologies, pp. 43–46. IEEE (2001)
10. Ortony, A., Clore, G.L., Collins, A.: The Cognitive Structure of Emotions. Cambridge University Press, Cambridge (1990)
11. Paquette, L., Baker, R.S.J.D., Sao Pedro, M.A., Gobert, J.D., Rossi, L., Nakama, A., Kauffman-Rogoff, Z.: Sensor-free affect detection for a simulation-based science inquiry learning environment. In: Trausan-Matu, S., Boyer, K.E., Crosby, M., Panourgia, K. (eds.) ITS 2014. LNCS, vol. 8474, pp. 1–10. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-07221-0_1
12. Picard, R.W., Papert, S., Bender, W., Blumberg, B., Breazeal, C., Cavallo, D., Machover, T., Resnick, M., Roy, D., Strohecker, C.: Affective learning - a manifesto. BT Technol. J. **22**(4), 253–269 (2004)
13. Russel, J.A.: A circumplex model of affect. J. Pers. Soc. Psychol. **39**, 1161–1178 (1980)
14. Shen, L., Leon, E., Callaghan, V., Shen, R.: Exploratory research on an affective e-learning model. In: Proceedings of Workshop on Blended Learning, pp. 267–278 (2007)

Module Advisor: Guiding Students with Recommendations

Nina Hagemann(✉), Michael P. O'Mahony, and Barry Smyth

School of Computer Science, Insight Centre for Data Analytics,
University College Dublin, Dublin, Ireland
nina.hagemann@insight-centre.org
http://www.insight-centre.org

Abstract. Personalised recommendations feature prominently in many aspects of our lives, from the movies we watch, to the news we read, and even the people we date. However, one area that is still relatively underdeveloped is the educational sector where recommender systems have the potential to help students to make informed choices about their learning pathways. We aim to improve the way students discover elective modules by using a hybrid recommender system that is specifically designed to help students to better explore available options. By combining notions of content-based similarity and diversity, based on structural information about the space of modules, we can improve the discoverability of long-tail options that may uniquely suit students' preferences and aspirations.

Keywords: Recommender systems · Content-based filtering
Diversity · Collaborative filtering · Module recommendations
Elective modules

1 Introduction

Today's students enjoy a wide variety of options regarding the availability of courses and modules, encouraging students to broaden their horizons, explore their interest and strengths, and develop new skills. One such opportunity offered in many universities is the possibility to freely choose elective modules from outside a student's main area of study. The taking of such elective modules is often a mandatory requirement of programmes of study, and can have a significant impact on students' academic experience and overall performance.

Unfortunately, in practice, student choices are often limited by discoverability challenges and overcrowded modules as students flock to popular options. As a result many students follow the crowd or their peers' recommendations when selecting electives. This trend was confirmed to exist in a preliminary exploratory data analysis of the Computer Science undergraduate students in our institution. An analysis of historical student data revealed an imbalance in elective module allocations. The percentage of students choosing modules outside of Computer

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 319–325, 2018.
https://doi.org/10.1007/978-3-319-91464-0_34

Science decreased rapidly over time; instead, students selected from a limited set of popular modules. This lead to many unsuccessful allocations (given constraints on enrolment numbers), obliging students to settle for their second or third elective module choices. We hypothesise that one of the reasons for these trends is the low discoverability of elective modules, especially those outside of a student's core area. Therefore, our main objective is to support students in discovering elective modules outside of their main field of study.

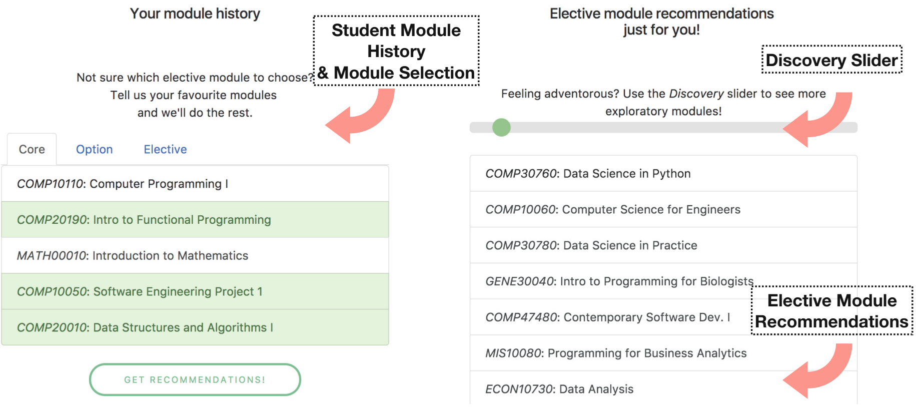
The need for a recommender system for academic guidance has been established over ten years ago. Previous research has shown the possibilities and requirements for such systems [2,4,11]. More recently, the interest in recommender systems for the educational sector has grown and studies agree on the benefit of recommender systems for module exploration [1,5]. However, the majority of this research focuses on grade prediction or the use of grades as an indirect way of measuring students' ratings of modules [3,6].

Although we agree that success in a module is an important factor for students to choose their modules, in this work we focus on the content of a module and its relevance to students' interests. We focus on supporting students in finding modules that are related to, but outside, their main area of study. We developed a prototype application that includes a personalised recommender system which helps students to discover lesser known elective modules by introducing diversity into the recommendation process.

In this paper we briefly present the current prototype and the underlying recommender system techniques and discuss the results of a preliminary offline study.

2 User Interface Prototype

To help students discover suitable elective modules we developed a prototype web application as shown in Fig. 1. The application includes a personalised



Your module history

Student Module History & Module Selection

Not sure which elective module to choose? Tell us your favourite modules and we'll do the rest.

COMP10110: Computer Programming I
COMP20190: Intro to Functional Programming
MATH00010: Introduction to Mathematics
COMP10050: Software Engineering Project 1
COMP20010: Data Structures and Algorithms I

Elective module recommendations just for you!

Discovery Slider

Feeling adventorous? Use the *Discovery* slider to see more exploratory modules!

COMP30760: Data Science in Python
COMP10060: Computer Science for Engineers
COMP30780: Data Science in Practice
GENE30040: Intro to Programming for Biologists
COMP47480: Contemporary Software Dev. I
MIS10080: Programming for Business Analytics
ECON10730: Data Analysis

Elective Module Recommendations

Fig. 1. Screenshot of the recommender system part of the prototype

recommender system where students can choose modules from their module history and receive elective module recommendations based on their choice. A slider allows students to control the degree of *discovery* in the recommendations that are made. Moving the slider introduces diversity into the recommender system algorithm and acts as a natural explanation for the recommended modules. Thus, students are facilitated to gradually explore modules outside of their field of study and to broaden their knowledge about available modules in different areas.

3 Module Recommendation Approaches

In this section, we describe the proposed hybrid approach to elective module recommendation used in the application. We further briefly describe a collaborative filtering approach that is used to evaluate our results. The following notation is introduced. Let S and M denote the set of students and modules in the system, respectively. Each student, $s_i \in S$, is profiled by a subset of the modules which they have previously taken. Let P_i denote the profile of student s_i, where $P_i = \{m_1, m_2, \ldots, m_l\}$ and $m_j \in M$ denotes a particular module. Based on the modules in the profile, candidate elective modules (i.e. all elective modules offered by the university) are ranked and a top-N list of recommendations is returned for student s_i.

The proposed hybrid recommender consists of two components to produce recommendations. The first component prioritises candidates that are similar in content to those in the student's profile; for this purpose, a traditional content-based (CB) recommender is used. The second component prioritises candidates from outside the student's programme area; in this case, a hierarchical taxonomy of the available programmes of study and associated modules is created, and candidates which are furthest from those in the student's profile are recommended.

Content-Based Recommender. Each module has a descriptor which provides a textual description of its content, aims and learning outcomes. Thus, modules can be viewed as documents made up of the set of terms contained in their descriptors. Using the Vector Space Model (VSM) [9], each module is represented by a vector in an n-dimensional space, where each dimension corresponds to a term from the overall set of terms in the collection. Standard preprocessing of documents is performed, such as tokenisation, stop-words removal, and stemming [7]. Each module is represented as a vector of term weights, where each weight indicates the degree of association between the module and the corresponding term. For term weighting, we employ TF-IDF (Term Frequency-Inverse Document Frequency) [8], a commonly used scheme in information retrieval. The intuition behind TF-IDF is that a term which occurs frequently in a given document (TF), but rarely in the rest of the collection (IDF), is more likely to be representative of that document. Given the vector space representation the similarity between two modules is computed using cosine similarity [8].

The rank score of a candidate elective module, m_c, for student s_i is calculated as the mean cosine similarity between m_c and each of the modules in the student's profile, P_i, as follows: $\text{score}_{CB}(s_i, m_c) = \frac{1}{|P_i|} \sum_{m_j \in P_i} \text{sim}(m_c, m_j)$. Candidates are ranked in descending order of score.

Taxonomy-Based Recommender. In order to recommend modules to students from outside their programme of study, an approach based on a hierarchical taxonomy of the academic structure of our university is used. Briefly, there are six Colleges, each with a number of constituent Schools. Each School offers a number of programmes of study, and each module is associated with one or more of these programmes.

While more sophisticated approaches are possible, here we make the general assumption that modules from the same programme are more closely related than those from different programmes. The following approach to used to calculate the rank score of a candidate elective module m_c for a given student s_i with profile P_i: $\text{score}_{TB}(s_i, m_c) = \frac{1}{|P_i|} \sum_{m_j \in P_i} \text{rel}(m_c, m_j)$, where $\text{rel}(m_c, m_j)$ is 0 if both modules belong to the same programme; 0.33 if the modules are from different programmes offered by the same School; 0.66 if the modules are offered by different Schools in the same College; and 1 if the modules are from programmes offered in different Colleges. Using this approach, higher scores are assigned to candidate modules which are further from those in the student's profile, thereby facilitating the student to broaden their learning experience.

Hybrid Recommendation Ranking. The above provides two alternatives to elective module recommendation. The former prioritises candidates which are similar to a student's profile, while the latter prioritises candidates which are furthest from a student's core programme of study. These approaches can be combined to allow students to better explore the wide range of elective module choices available from across the university. An overall score for a candidate elective module m_c is calculated for student s_i as follows: $\text{score}(s_i, m_c) = \alpha \, \text{score}_{CB}(s_i, m_c) + (1 - \alpha) \, \text{score}_{TB}(s_i, m_c)$, where the parameter α can be varied to influence the diversity of elective modules recommended.

Collaborative Recommender. We also consider a neighbourhood-based *collaborative filtering* (CF) approach [12]. As before, each student s_i is profiled by a subset of previously taken modules, P_i. The neighbourhood for a given student s_i is determined based on profile similarity, where the similarity between two profiles, P_i and P_j, is calculated using the overlap coefficient [13]. Once the k most similar students to student s_i are identified, a top-N list of elective module recommendations, ranked by their frequency of occurrence in neighbour profiles, is then returned to the student. Using this approach, the elective modules which are popular among students with similar profiles are recommended.

4 Evaluation

We randomly selected 100 Computer Science students from the historical data set. Each student is represented by an average of 20 core modules, from which we randomly select three as the input to the recommender system, mimicking a student's input into the web application.

We conduct a leave-one-out test [10] and generate a top-10 recommendation set for each student for each recommendation approach: a pure content-based approach ($\alpha = 1$), three hybrid approaches ($\alpha = [0.25, 0.5, 0.75]$), and the collaborative filtering method (CF). To evaluate the offline results we are not using a classic accuracy score as we hypothesise that our ground truth, that is the set of elective modules actually taken by students, is skewed due to the reasons explained above (i.e. students largely following peer recommendations or simply choosing popular modules). One of our main objectives is to broaden the range of modules that students are aware of. Hence, we evaluate our results comparing the number of distinct modules recommended over all users, and the number of distinct subjects covered by these recommendations. To evaluate relevance, we use *sim-to-core* (*StC*), a metric that determines the average similarity of the most similar module in the student's profile, P_i, to each module in the recommendation set, R_i, as shown in Eq. 1:

$$StC(P_i, R_i) = \frac{\sum_{m_k \in R_i} sim_{max}(m_k, P_i)}{|R_i|}, \tag{1}$$

where $R_i = \{m_1, ..., m_r\}$ is the set of elective module recommendations and $sim_{max}(m_k, P_i)$ returns the maximum similarity between the recommended elective module m_k and the modules in the student's profile.

4.1 Results and Discussion

Firstly, we consider the overlap coefficient [13] of the recommendation sets produced by the various approaches (Table 1). As expected, as more diversity is introduced into the recommendation process (i.e. as α is decreased), a decrease in overlap between the recommended sets is observed. For example, an overlap of 76.5% in recommended sets is seen between the pure content-based recommender ($\alpha = 1$) and the hybrid approach with $\alpha = 0.5$. Comparing the recommendations made by the collaborative filtering approach, we see approximately only 3% of the same modules being recommended; since this approach operates over the limited set of largely popular modules actually selected by students, this result is also to be expected.

Table 2 shows that there is a gradual increase in both the number of distinct modules (D. Mod.) recommended and the number of distinct subjects covered (D. Sub.) as diversity is introduced (i.e. as α decreases). Moreover, the percentage of in-programme (Computer Science) modules (% IPE) recommended also reduces, while the reduction in the *sim-to-core* (*StC*) metric is less pronounced. The results also show that the collaborative filtering approach produces recommendations with the lowest number of distinct modules and subjects covered, while the percentage of IPE modules recommended is the highest. Thus, it can be seen that the hybrid approach can successfully improve recommendation diversity, without significantly compromising relevance, while the collaborative filtering approach recommends from a relatively small set of modules.

Table 1. Overlap of the recommended module sets by the different approaches.

α	1	0.75	0.5	0.25	CF
1	1.000	0.901	0.765	0.626	0.031
0.75		1.000	0.862	0.724	0.032
0.5			1.000	0.860	0.033
0.25				1.000	0.034
CF					1.000

Table 2. Evaluation results for the different approaches.

α	D. Mod.	D. Sub.	% IPE	*StC*
1	149	31	24.1	0.012
0.75	156	34	21.1	0.011
0.5	157	37	17.9	0.009
0.25	154	44	12.3	0.008
CF	60	28	26.7	0.002

5 Conclusion and Future Work

The application of recommender systems seems opportune given the increasing tendency of our university's students to select from among a limited number of popular elective modules. We have shown that module descriptions can be used to make meaningful recommendations. While collaborative filtering approaches will give accurate results in a traditional sense, it will not help the problem of discoverability of modules as it promotes primarily already popular modules. We have shown that the proposed hybrid recommender system can add diversity to the set of recommendations. While the taxonomy-based recommender represents a first step, nonetheless it is capable of facilitating the discoverability of modules outside of the students' core area of study. In future work we plan on further developing our approach as well as conducting a live user study to understand how students will interact with the system and whether it leads to students choosing from among a more diverse range of elective module options.

References

1. Ajanovski, V.V.: Guided exploration of the domain space of study programs. In: 4th Joint Workshop on Interfaces and Human Decision Making for Recommender Systems (IntRS) 2017, p. 43 (2017)
2. Bendakir, N., Aïmeur, E.: Using association rules for course recommendation. In: Proceedings of the AAAI Workshop on Educational Data Mining, vol. 3 (2006)
3. Bydžovská, H.: Are collaborative filtering methods suitable for student performance prediction? In: Pereira, F., Machado, P., Costa, E., Cardoso, A. (eds.) EPIA 2015. LNCS (LNAI), vol. 9273, pp. 425–430. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-23485-4_42
4. O'Mahony, M.P., Smyth, B.: A recommender system for on-line course enrolment: an initial study. In: Proceedings of the 2007 ACM Conference on Recommender Systems, RecSys 2007, pp. 133–136. ACM, New York (2007)
5. Park, Y.: A recommender system for personalized exploration of majors, minors, and concentrations. In: CEUR Workshop Proceedings 1905 (2017)

6. Polyzou, A., Karypis, G.: Grade prediction with course and student specific models. In: Bailey, J., Khan, L., Washio, T., Dobbie, G., Huang, J.Z., Wang, R. (eds.) PAKDD 2016. LNCS (LNAI), vol. 9651, pp. 89–101. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-31753-3_8
7. Porter, M.F.: An algorithm for suffix stripping. In: Readings in Information Retrieval, pp. 313–316. Morgan Kaufmann Publishers Inc., San Francisco (1997)
8. Salton, G., McGill, J.: Introduction to Modern Information Retrieval. McGraw-Hill Inc., New York (1986)
9. Salton, G., Wong, A., Yang, C.: A vector space model for automatic indexing. Commun. ACM **18**(11), 613–620 (1975)
10. Sammut, C., Webb, G.I.: Leave-one-out cross-validation. In: Sammut, C., Webb, G.I. (eds.) Encyclopedia of Machine Learning, pp. 600–601. Springer, Boston (2010). https://doi.org/10.1007/978-0-387-30164-8_469
11. Sandvig, J., Burke, R.: AACORN: a CBR recommender for academic advising. Technical report, TR05-015, DePaul University (2005)
12. Shardanand, U., Maes, P.: Social information filtering: algorithms for automating "word of mouth". In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI 1995, pp. 210–217. ACM Press/Addison-Wesley Publishing Co., New York (1995)
13. Vijaymeena, M., Kavitha, K.: A survey on similarity measures in text mining. Mach. Learn. Appl.: Int. J. **3**(2), 19–28 (2016)

TUMA: Towards an Intelligent Tutoring System for Manual-Procedural Activities

Zardosht Hodaie(✉), Juan Haladjian, and Bernd Bruegge

Chair for Applied Software Engineering,
Technische Universität München, Munich, Germany
{hodaie,haladjian,bruegge}@in.tum.de

Abstract. Many activities, such as learning a craft, involve learning how to manipulate physical objects by following a step-by-step procedure. In this paper we present our ongoing work on development of TUMA: an intelligent tutoring system for manual-procedural activities. We first introduce the notion of *manual-procedural activity* and then argue about the opportunities for creating intelligent tutors for manual-procedural activities. Such an intelligent tutoring system can be used in domains like teaching crafts, that involve acquiring cognitive knowledge along with specific motor skills. TUMA unifies the research from three different communities: intelligent tutoring systems, human motion tracking, and assistance systems for manual assembly in manufacturing. We describe the vision and the requirements of TUMA and its functional architecture inspired by high-level components of intelligent tutoring systems. Finally we report on our research road map for implementing a proof-of-concept and evaluating its impact.

Keywords: Intelligent tutoring systems
Manual-procedural activities · Hand tracking · Skill acquisition
Activity tracking · Manual assembly · Crafts training

1 Introduction

Many activities, such as crafts and manual work in factories, involve learning a step-by-step procedure for manipulating physical objects. Common to performing all these activities is (a) manipulating physical objects either directly by hands, or indirectly using tools; and (b) following steps of a given procedure, either by looking it up in the instruction manual, or knowing it by heart, or imitating an instructor. We propose the notion of a *manual-procedural activity (MPA)* to distinguish these kinds of activities.

A manual-procedural activity consists of multiple steps that must be performed in a specific order, and each steps involves manipulating physical objects, possibly by using tools. activities. Learning a MPA involves acquiring procedural knowledge about the steps of a procedure and the domain of the task, as well as motor skills for manipulating objects and using tools. We can consider

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 326–331, 2018.
https://doi.org/10.1007/978-3-319-91464-0_35

a spectrum for classification of MPAs based on two dimensions: (a) how clear are the step boundaries, (b) how complex are the required hand skills (Fig. 1, left). This classification helps us better analyze the requirements of an ITS for manual-procedural activities.

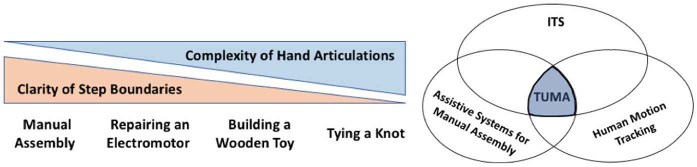

Complexity of Hand Articulations
Clarity of Step Boundaries
Manual Assembly
Repairing an Electromotor
Building a Wooden Toy
Tying a Knot
ITS
TUMA
Assistive Systems for Manual Assembly
Human Motion Tracking

Fig. 1. Left: the spectrum of manual-procedural activities with examples. Right: TUMA as intersection of three research areas.

The goal of TUMA, an intelligent tutoring system for manual-procedural activities, is to help workers and trainees learn a manual-procedural activity. The vision is to develop a system that acts as a trainer in learning of a manual-procedural activity. Similar to a human trainer, our system should guide the apprentice through the learning of a craft by telling him how to do the steps towards a goal, track his performance, and give him individualized feedback on his mistakes, until a desired level of mastery in both procedural and motor skills is achieved. We limit the scope of the project to those activities that are performed on a workbench.

In the rest of this paper we first argue that by unifying the research findings from three communities, intelligent tutoring systems (ITS), assistive systems in manufacturing, and human motion tracking, it is possible to build an intelligent tutoring system for manual-procedural activities. We then propose a functional architecture of such an ITS and explain our research roadmap for measuring its effectiveness in learning different MPAs.

2 Background and Related Work

Bloom's taxonomy [7], categorizes the learning objectives into three main domains of cognitive, affective, and psychomotor. The cognitive domain involves knowledge and targets acquiring mental skills required to solve intellectual problems, like solving a mathematical equation. The affective domain involves emotions and targets acquiring skills that shape feelings, values, motivations, and attitudes. The psychomotor domain involves physical movements of the body and motor and coordination skills.

Learning a MPA involves both cognitive and psychomotor domains. On the one hand, the trainee should learn about the steps of procedure, the objects, materials, and tools involved, and the theories behind the actions. On the other hand, he should acquire the required motor skills to use tools and manipulate objects. In the following, we provide a brief overview of three research areas that are relevant for teaching the cognitive and psychomotor skills of a manual-procedural activities: intelligent tutoring systems, human motion tracking, and assistance systems for manual assembly. Each of these areas contributes to parts of the requirements of an ITS for MPA. We consider therefor the TUMA as an integrating work at the intersection of these research areas (Fig. 1, right).

There is a large body of research on computer-based training and Intelligent Tutoring Systems (ITS). An ITS supports learning by providing instructions and real-time personalized feedback to the students by applying different pedagogical methods and following different pedagogical strategies [5]. An ITS typically has four main components: *domain model* contains the information about the domain of the task; the *student module* is responsible for tracking individual performance of the student and providing individually tailored instructions and feedback; the *pedagogical module* contains models of different pedagogical strategies and methods and is responsible for guiding the learning path by selecting the appropriate instruction and exercise; and the *communication module* is the interface between the student and the ITS and is responsible for collecting input and presenting the instructions, exercises, and feedback to the student. The research on ITSs has so far mainly focused on the cognitive domain of learning and the psychomotor domain has been mostly neglected [2,6]. This might be due to the challenges involved in tracking and interpreting body movements in a format that is appropriate for the reasoning of an ITS.

Nevertheless, we argue that the advances in the state of the art in human motion tracking using computer vision and body-attached motion sensors provide the opportunity for creating ITSs that address psychomotor learning. Currently, there exists a large body of work on tracking physical movements of humans for different purposes, such as rehabilitation, measuring the performance, feedback on body posture, learning sign language [1,4]. These works however often do not consider the pedagogical advantages of intelligent tutoring systems, specifically the adaptive, personalized feedback [6].

Assistance systems in industrial settings, specially for manual assembly, is another area of research that has focused on guiding a worker through a complex manual procedure [3]. These systems also often use simple form of activity tracking, but their focus is mostly on guiding the worker through the process and do not consider pedagogical aspects of learning the process or the psychomotor skills. Furthermore, the principles of assembly-oriented design suggest designing products in way that require least degree of complex hand movements.

Our proposed system is inspired by and overlaps with research in these three areas. Applying pedagogical methods and strategies, tracking individual performance of trainees and providing individual feedback and instructions is inspired by intelligent tutoring systems. The guidance and the applications of the system, specially for manual-procedural activities with clear step boundaries and simple hand articulations leans on the research in the assistance systems for manual assembly. And finally, teaching complex hand articulations and motor skills is based on the state of the art in vision-based and body-sensor-based human motion tracking.

3 TUMA: An Intelligent Tutoring System for Manual-Procedural Activities

At the highest level, an ITS for manual-procedural activities should provide:

- guidance and step-by-step walk-through of a procedure (similar to assistance systems in manual assembly),
- tracking of the hand movements, detecting the gestures and tool usages and evaluating their quality (similar to human motion tracking), and
- detection and explanation of mistakes and adaptive personalized instructions and feedback to the learner by following different pedagogical strategies (similar to ITS).

A major challenge in development of ITSs is authoring. TUMA addresses the problem of authoring of the ITS for manual-procedural activities by applying a Programming-by-Demonstration approach. Clearly, the domain model and the pedagogical component need to be authored separately. However, the expert model of the hand movements is recorded using the tracking component and used later as the reference to determine the quality of trainee's movement. Furthermore, for MPAs that have a more clear step boundaries, the steps of the process and the objects and tools involved in each step can also be captured automatically. The expert can then edit and enhance this automatically captured workflow for guiding trainees through the process.

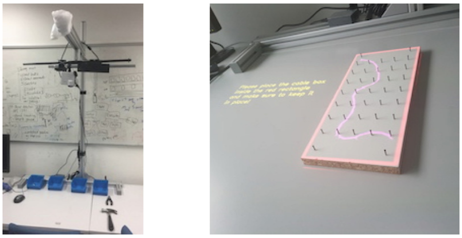


Fig. 2. Left: camera-projector setup with RGB-D camera. Right: step instructions are projected on the table.

TUMA should provide two working modes. In *execution mode* the trainee learns the MPA by performing it as the system guides him through the activity and gives personalized adaptive feedback on his the performance. In the *authoring mode* the system tracks and records the performance of the activity as the trainer demonstrates it. This mode involves creation of the expert model of performing the task as well as model of the expert hand movements in psychomotor phase of the task. What has been recorded during the authoring mode, i.e. the steps of the procedure, the object manipulations during each step, and the hand articulations involved in the performance of each step, will be used in

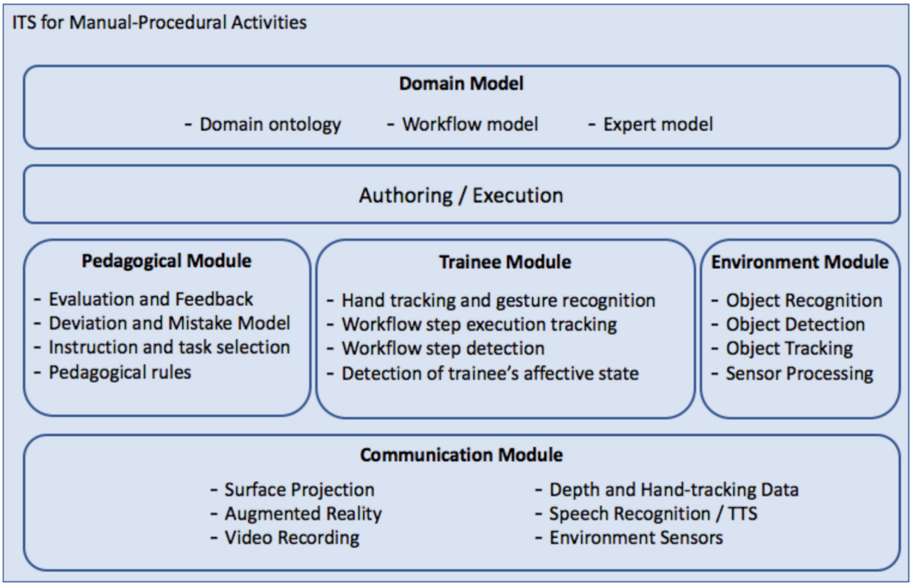


Fig. 3. Components of TUMA.

the execution mode to compare actions of the trainee with the model of expert actions and to provide feedback to the trainee based on this comparison.

To guide the trainees through a manual procedure that involved physical objects, TUMA should support multi-modal natural user interaction. We use a projector-camera setup for realization of TUMA (Fig. 2). The depth camera allows for tracking the hand movements. The instructions for each step and the feedback information are projected on the table. Speech interaction as well as motion sensor based tracking of hand movements will be added to this setup in next iterations of the project. Figure 3 shows the components of TUMA adapting the main components of an ITS. Considering the physical nature of an MPA, we need to extend the general model of the ITS with a component that is responsible for monitoring and perception of the user's environment. We call this component the *Environment Module*. The Environment Module is mainly responsible for detection and tracking of objects manipulated and tools used during a step of the workflow. Additionally, depending on the task domain this module is responsible for sensing different parameters of the environment, e.g. reaching a target temperature.

4 Research Roadmap

The evaluation of TUMA is formed around the three main questions:

(a) To what extent can we build a system that supports learning of manual-procedural activities, adhering to the pedagogical requirements?

(b) How effective is such a system in achieving the pedagogical goals of learning manual- procedural activities?
(c) To what extent can we facilitate as well as automate the process of authoring of instructional content for manual-procedural activities?

To the first question, we first collect the pedagogical requirements of learning manual- procedural activities. Beside reviewing of the related literature, we intent to do field observation and interviews with trainees and training personnel in the institutions where these trainings happen, such as factories and institutions for craftsmanship training. Based on the collected requirements we build the system and evaluate it against the collected set of the requirements.

To the second question, we focus our evaluation on three aspects of performance, retention of knowledge, and transfer of learning. In different case studies (e.g. manual assembly, cooking, crafts) we show the effectiveness of the system with regard to these three dimensions by comparing the effect of learning using TUMA with paper and video based baselines with regard to different metrics such as time to accomplish the task and number of errors.

Towards the third we evaluate to what extend the automated tracking and recording of the expert's actions (i.e. automated tracking of hand movements, object manipulations, tool usage, as well as the extracted sequence of actions) are enough and effective for authoring manual-procedural activities. To this end, we evaluate the system by applying the recorded MPAs to the execution mode of TUMA and measure the learning effect.

References

1. Filippeschi, A., Schmitz, N., Miezal, M., Bleser, G., Ruffaldi, E., Stricker, D.: Survey of motion tracking methods based on inertial sensors: a focus on upper limb human motion. Sensors **17**(6), 1257 (2017)
2. Goldberg, B.: Intelligent tutoring gets physical: coaching the physical learner by modeling the physical world. In: Schmorrow, D.D.D., Fidopiastis, C.M.M. (eds.) AC 2016 Part II. LNCS (LNAI), vol. 9744, pp. 13–22. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39952-2_2
3. Hinrichsen, S., Riediger, D., Unrau, A.: Assistance systems in manual assembly. In: Villmer, F.J. (ed.) Proceedings 6th International Conference on Production Engineering and Management, pp. 3–13 (2016)
4. Lun, R., Zhao, W.: A survey of applications and human motion recognition with microsoft kinect. Int. J. Pattern Recognit. Artif. Intell. **29**(05), 1555008 (2015)
5. Paviotti, G., Rossi, P.G., Zarka, D.: Intelligent tutoring systems: an overview. Pensa Multimedia (2012)
6. Santos, O.C.: Training the body: the potential of aied to support personalized motor skills learning. Int. J. Artif. Intell. Educ. **26**(2), 730–755 (2016)
7. Simpson, E.J.: The classification of educational objectives, psychomotor domain. Education, and Welfare, Office of Edcn, Department of Health (1970)

SAT Reading Analysis Using Eye-Gaze Tracking Technology and Machine Learning

Andrew Howe[1] and Phong Nguyen[2](✉)

[1] The American School in Japan, Tokyo, 106-0047, Japan
19howea@asij.ac.jp
[2] Tokyo Techies, Tokyo, 106-0031, Japan
phong.nguyen@tokyotechies.com

Abstract. We propose a method using eye-gaze tracking technology and machine learning for the analysis of the reading section of the Scholastic Aptitude Test (SAT). An eye-gaze tracking device tracks where the reader is looking on the screen and provides the coordinates of the gaze. This collected data allows us to analyze the reading patterns of test takers and discover what features enable test takers to score higher. Using a machine learning approach, we found that the time spent on the passage at the beginning of the test (in minutes), number of times switching between the passage and the questions, and the total time spent doing the reading test (in minutes) have the greatest impact in distinguishing higher scores from lower scores.

Keywords: Eye-gaze tracking · SAT reading · Machine learning · Analysis Reading pattern

1 Introduction

The SAT is a standardized test that American students choose to take in order to enroll in some of the more prestigious universities. Students need to know how to perform well; however, currently, there are no non-intrusive methods to help establish this knowledge. Previous analysis of the SAT reading section has utilized post-test surveys, which are highly subjective and depend completely on the students' biased answers [1]. Therefore, a more objective analysis of the SAT reading section needs to be conducted in order to provide students with the most reliable techniques for reading the passages. Inexpensive and accurate eye-gaze tracking technology, like the Ey-Tribe eye tracker, has recently become readily available to the general public. Data collection using this technology is simple, non-intrusive, and objective. Using this technology and Khan Academy official SAT practice tests, we were able to collect data whilst maintaining a natural test-taking environment. Data collected from the eye tracking device allows us to analyze and identify different reading patterns, as well as uncover features that have a significant impact on scores of test takers. After the collection of the initial data, we bifurcated the dataset into students who scored greater than or equal to 9, and students who scored less than 9, in order to identify any statistically significant habits in each respective group. With our new

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 332–338, 2018.
https://doi.org/10.1007/978-3-319-91464-0_36

bifurcated dataset, we built and trained a model to predict whether a student can achieve a score of 90% or above on a passage. Our predictive model can be applied to any data set gathered using our method and can be used as a tool to allow educators to correct students' behaviors in order to maximize test scores prior to taking the official SAT.

2 Related Work

Traditionally, College Board collects data about the SAT using qualitative post-test surveys. This has created a very unreliable and hard-to-analyze data set, as these post-test surveys are subjective and do not provide individualized feedback that the student can use to improve. Currently, post-test surveys are only used by College Board to improve the test or test-taking experience, and not student performance.

One method to reduce bias is to use a device that will collect data automatically and non-intrusively. Extensive research has been performed to verify the accuracy of eye-gaze tracking devices with respect to new sources of data including [2, 3]. In [2], Ooms et al. the experimenters verified that low-cost eye-gaze tracking devices can have comparable precision with the most well-established devices, given the correct setup and choice of software. Moreover, there is research in both [3, 4] that has proved that eye-gaze patterns are related with reading comprehension; however, the papers, respectively, lack a real-life application and deeper analysis.

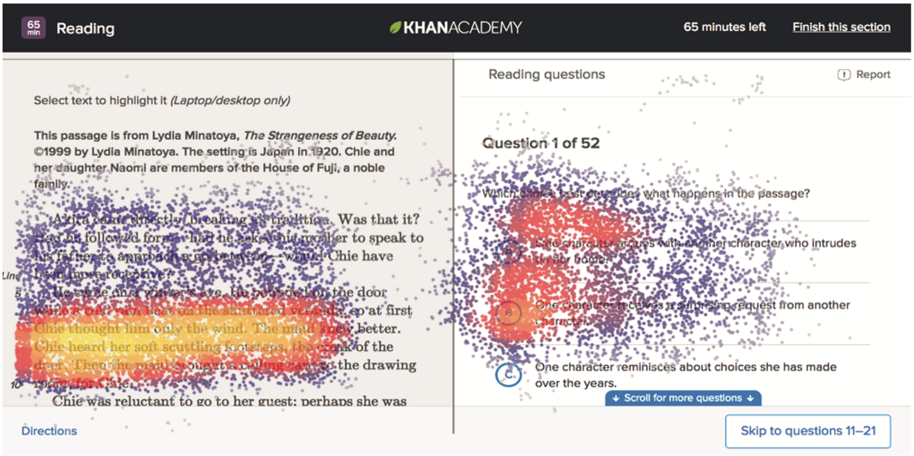
Our methodology aims to give students a more comprehensive and detailed way to further improve their test scores. In our previous paper [4] we described a method using correlation coefficients to analyze how reading pattern features correlate with scores. However, a major weakness in the previous method was that the correlation coefficient method was unable to show the difference between the reading methods of high achieving students and those of low achieving students. In addition, the correlation coefficient method does not show causality, and thus fails to achieve further insight into reading pattern behaviors. In this paper, we present a new method to further analyze the dataset using statistical tests and machine learning models. Our proposed method can show the significant difference in reading pattern features between students who have higher scores and those who do not and our machine learning model can give insight into how to achieve the best possible scores.

3 Proposed Method

This experiment used eye-gaze coordination data points in the form of time-series data in order to infer the reading patterns of SAT test takers. From the reading pattern features, we built a machine learning model to predict the outcome of the test. We interpreted that model and found the relationship between the reading patterns of the SAT test takers and their results. In the following subsection, we will explain each step we performed in order to achieve our results, from data collection to the creation of our prediction model.

3.1 Data Collection Method

We used Khan Academy's official practice SAT as the environment [5], where the passage was on the left side and the questions are on the right side of the screen. Students were required to take the test online while the eye-gaze tracking device was collecting the data. The coordinates of where the students were looking were stored in a CSV file for analysis. We started collecting the data when the student began the test and we ended the collection process after the student answered all of the questions pertaining to a single passage. Figure 1 shows the visualization of a test taker's heat map of eye gaze on the Khan Academy interface.



Select text to highlight it *(Laptop/desktop only)*

This passage is from Lydia Minatoya, *The Strangeness of Beauty.* ©1999 by Lydia Minatoya. The setting is Japan in 1920. Chie and her daughter Naomi are members of the House of Fuji, a noble family.

Akira came directly, breaking all tradition. Was that it? Had he followed form—had he asked his mother to speak to his father to approach a go-between—would Chie have been more receptive?

He came on a winter's eve. He pounded on the door while a cold rain beat on the shuttered veranda, so at first Chie thought him only the wind. The maid knew better. Chie heard her soft scuttling footsteps, the creak of the door. Then the maid brought a calling card to the drawing room, for Chie.

Chie was reluctant to go to her guest; perhaps she was

Question 1 of 52

Which choice best describes what happens in the passage?

A One character argues with another character who intrudes on her home.

B One character receives a surprising request from another character.

C One character reminisces about choices she has made over the years.

Fig. 1. Heat map of eye gazes on Khan Academy's SAT passage test interface

In addition, we manually took note of the number of correct answers, incorrect answers, and unanswered questions as the ground truth information of the analysis.

3.2 Feature Extraction

Collected eye-gaze data were in a time-series format. From each sequence, we extracted 12 features from the raw coordinates of eye-gazes [4]. Table 1 below lists the 12 features.

Table 1. List of reading pattern features and their explanation

No.	Feature name	Explanation
1	Total time	Total time to complete one passage (in minutes)
2	Percentage of total time looking at the passage	Percentage of total time looking specifically at the passage (in percent)
3	Percentage of total time looking at the questions	Percentage of total time looking specifically at the questions (in percent)
4	Total switch count	Total amount of times the subject switches from looking at the passage to looking at the questions or vice versa (in times)
5	Time passage beginning	Amount of time spent solely on reading the passage at the beginning of the test (in minutes)
6	10-sec intervals passage reading	Number of ten-second intervals which have 80% or more of the ten seconds spent on reading for the passage (in times)
7	10-sec intervals question reading	Number of ten-second intervals which have 80% or more of the ten seconds spent on reading for the questions (in times)
8	Percentage of time looking at the passage in first 4 min	Percentage of time in first four minutes spent only on reading the passage (in percent)
9	Percentage of time looking at the questions in first 4 min	Percentage of time in first four minutes spent only on reading the question (in percent)
10	Percentage of time looking at the passage in last 4 min	Percentage of time in last four minutes spent only on reading the passage (in percent)
11	Percentage of time looking at the questions in last 4 min	Percentage of time in last four minutes spent only on reading the question (in percent)
12	Speed of reading the passage	The speed at which the test taker reads the passage measured by the distance between two pixels in a fixed interval

3.3 Machine Learning Model

Typically, a machine learning model is trained from past example data and has the capability to predict an unseen dataset. However, we had a different approach. We trained a machine learning model and analyzed how the model learns to make the decisions in order to extract insights on the model. We split the data into two groups based on the results of the reading test: those that answered correctly 9 or 10 out of 10 questions (high score), and those that answered correctly lower than 9 out of 10 questions (low score). We chose a simple decision tree classifier to train a classification model. The decision tree used the above features and learned how to determine whether the outcome of the test would result in a high score or low score. We used the whole dataset for training because our purpose was to extract the decision processes of the model not predict future occurrences. We also reported how well the model can perform on the same training dataset.

4 Experimental Settings

We used an EyeTribe as our eye-gaze tracking device. The EyeTribe's sampling rate is 30 Hz. On average, for each reading test with one passage, we had around 12,000 coordinates in total.

Our test takers ranged in age from 14 to 18 years old: the ultimate beneficiary ages of this experiment. We asked them to read a passage and answer all the questions on the screen with no guidance on the reading method. After the first test, the students took the test two more times, but with a certain level of guidance. The first time, they were asked to read the passage first, and then move to the questions. The second time, they were asked to do the same method but in reverse order.

We collected a dataset of 30 native English speaking students from 10 different countries. In total, we had over 1000 min of reading data on 90 reading tests. Each test was associated with a particular score that they achieved for that test, ranging from 0 to 10.

5 Results

After training the decision tree model to classify whether a test was a high score or low score, we realized that the decision tree had stopped using more features after four splits. Our decision tree used entropy as the criterion for measuring the quality of a split. The three features used by the decision tree were time passage beginning, total time, and total switch count. The feature at the root of the decision tree had the highest information gain. The decision tree is illustrated in Fig. 2.

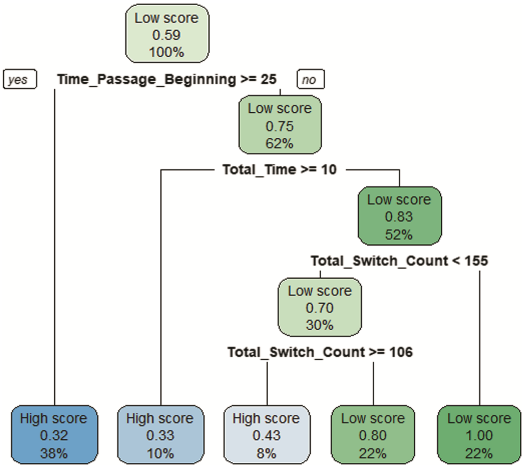


Fig. 2. Decision tree classification model decision processes

If the test taker reads the passage for 25 s or more at the beginning of the test, he or she will be most likely to score higher (23 cases out of 34 cases). For those who read the passage for less than 25 s at the beginning of the test and have his or her total time either equal to 10.49 or be greater than 10.49 min, he or she will most likely score higher (6 cases out of 9 cases). A student who switches fewer than 155 times but more than 106 times between the passage and the questions will also be more likely to have a higher score.

The decision tree had an accuracy of 76.67% predicting its own training dataset. The confusion matrix of the decision tree is described in Table 2 below.

Table 2. Confusion matrix of the decision tree classification model

Prediction	Ground truth	
	Higher-score	Lower-score
Higher-score	**33**	17
Lower-score	4	**36**

According to the results, we can infer some of the following insights: (i) A test taker should spend 25 s or more reading the passage at the beginning; (ii) a test taker should spend as much time as possible on a single passage (10.49 min or more), but pace themselves efficiently because one is only allotted 65 min for 52 questions; and (iii) a test taker should not switch between the questions and the passage too many times.

6 Conclusion

We have proposed a method to analyze the reading patterns of SAT reading test takers from raw eye-gaze tracking data. From the analysis, we can extract the reading pattern features and use them in a machine learning model. The machine learning model we chose is the decision tree classification model. This model gives us insights on how to improve student performance on the reading section of the SAT. Based on the results of the trained model, the features such as spending time on reading the passage at the beginning, the total time to take the test, and the number of switches between the questions and the passage are found to have a significant impact on SAT reading performance.

We plan to collect a bigger dataset in order to expand the research and use the predictive model in a more real-life application to help improve SAT reading test performances.

References

1. “New SAT Survey Press Release.” New SAT Survey Press Release, Public Relations The Princeton Review. The Princeton Review, Princeton Review
2. Ooms, K., Dupont, L., Lapon, L., Popelka, S.: Accuracy and precision of fixation locations recorded with the low-cost Eye Tribe tracker in different experimental set - ups. J. Eye Mov. Res. **8**, 1–24 (2015). https://doi.org/10.16910/jemr.8.1.5

3. Vo, T., Mendis, B.S.U., Gedeon, T.: Gaze pattern and reading comprehension. In: Wong, K.W., Mendis, B.S.U., Bouzerdoum, A. (eds.) ICONIP 2010. LNCS, vol. 6444, pp. 124–131. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-17534-3_16
4. Andrew, H., Phong, N.: Open source software for analysis and correlation of reading patterns with superior SAT scores using gaze-tracking device. In: Proceedings of IETC Proceedings Book, pp. 50–56 (2017)
5. Khan Academy. https://www.khanacademy.org/mission/sat/exams

Determining What the Student Understands - Assessment in an Unscaffolded Environment

C. W. Liew(✉) and H. Nguyen

Lafayette College, Easton, PA 18042, USA
{liewc,nguyenha}@lafayette.edu

Abstract. Assessment of skills and process knowledge is difficult and quite different from assessing knowledge of content. Many assessment systems use either multiple choice questions or other frameworks that provide a significant amount of scaffolding and this can influence the results. One reason for this is that they are easy to administer and the answers can be automatically graded. This paper describes an assessment tool that does not provide scaffolding (and therefore hints) and yet is able to automatically grade the free form answers through the use of domain knowledge heuristics. The tool has been developed for a tutoring system in the domain of red black trees (a data structure in computer science) and has been evaluated on three semesters of students in a computer science course.

Keywords: Assessment · Computer science
Unscaffolded environments

1 Introduction

One of the keys to developing effective tutoring systems is the ability to determine what a students knows or understands both before and after a tutoring session. Without this capability, we would be unable to either develop effective tutoring strategies or to evaluate the efficacy of any teaching or tutoring approach. However assessing a student's knowledge, skills or conceptual understanding can be a difficult task. It's not enough to determine what they can or cannot do but if we are to help students improve, we have to be able to narrow down to the specific issues that are a problem and also the contexts in which they occur.

This paper describes a tool that accurately assesses student knowledge and skills without needing any scaffolding that would bias the assessment. The tool consists of two modules that provide a non-scaffolded test taking environment and corrects the student answers to the tests. The grading module identifies (1) correct and incorrect answers, (2) where the first error occurs in incorrect answers and (3) the type of error in the majority of the cases. The tool and the

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 339–344, 2018.
https://doi.org/10.1007/978-3-319-91464-0_37

associated tutoring system have been implemented for the domain of balanced binary trees, an important data structure in computer science. The tool has been successfully used to assess three semesters of students taking the data structures class at our institution.

2 Problem Domain: Balanced Binary Trees - Red Black Trees

Binary trees are a fundamental data structure in computer science and are commonly taught in the second computer science course (CS2). Balanced binary trees are a natural extension of binary trees and provide good performance for sorting and searching regardless of how the input data is arranged. Red black trees are one particular type of balanced binary trees - others include AVL [4] trees and 2–3 trees [1]. With data structures like these, students have to know how to program them (use them) also have to understand the underlying concepts underlying the operations. The basic insertion and deletion operations are much more complex when compared to those of a standard binary tree.

A red black tree is a self balancing binary search tree that has the following properties [5]:

1. The nodes of the tree are colored either red or black.
2. The root of the tree is always black.
3. A red node cannot have any red children.
4. Every path from the root to a null link contains the same number of black nodes.

The top-down algorithms [5] to insert or delete a value from a red-black tree starts at the root and, at every iteration, moves down to the next node, which is a child of the current node. At each node, it applies one or more transformation rules so that when the actual insertion (or deletion) is performed no subsequent actions are needed to maintain the tree's properties. Other types of balanced trees also use a similar approach. In our work we used red-black tree as an exemplar to evaluate our ideas and implementations, but they should be applicable to balanced trees in general. The transformation rules for insertion are called *color flip*, *single rotation*, *double rotation*, *simple insertion*, and *color root black*. There are more rules for deletion and they are called *color flip*, *single rotation*, *double rotation*, *simple deletion*, *switch value*, *drop*, *drop&rotate* and *color root black*.

3 Assessment and Related Work

Many tutoring systems have designed tests that accept answers in a restricted input language (e.g., numerics only). This leaves the problem with a large solution space that renders guessing ineffective. Previous work on developing assessment tools in unscaffolded environments used domain knowledge to infer structure and reasoning. For example, the PHYSICS-TUTOR system [2] used knowledge of basic physics to heuristically determine the intent of students when they are solving mechanics problems in introductory physics courses. They used domain knowledge, algebraic knowledge and heuristics to accurately determine correctness and errors even if there were missing equations or factors and the answers contained numbers instead of variables.

In the red black tree domain students are assessed not for their knowledge of content but their skill in applying the insertion and deletion algorithms both of which require that a student knows how to apply the transformations and can recognize when they are applicable. In addition, students can either combine several base (canonical) steps into a larger one or can change the order in which transformations are applied (typically with a *color flip*). Multiple choice questions are not a good mechanism for assessing skills in this domain. The ideal assessment tool would handle these issues and provide an environment similar to a test where students are provided with a blank sheet of paper and free response questions. The questions would test the students understanding and the tool would allow the students to make the same mistakes that they would make on the paper test. The tool would also be able to grade the exam and determine each student's problem areas.

4 The Tutoring and Assessment Interfaces

The system has 3 sections - the pre-test, the tutor, and the post-test. In the test sections, a typical insertion (deletion) problem for red-black trees involves inserting a sequence of numbers to a starting tree (or deleting from it). Students have to show the state of the tree after every insertion/deletion; they are also encouraged to show any intermediate states (the trees that are created along the path to the solution). To this end, the test interface displays a "blank" binary tree canvas of 31 empty nodes. The student can click on any node to specify its value and color - submitting a tree is therefore equivalent to placing all of its nodes in the appropriate position in the tree canvas; nodes that are left empty are assumed to be null black nodes. The interface is designed to look like a sheet of paper with blanks to fill in - in this way, we ensure that the system does not provide any hints or clues as to what the desired answer would be.

In the tutoring section, students perform the same task of inserting to (or deleting from) a starting tree. However, a node-by-node modification of the current tree is not required; instead, students only need to select a node and the transformation to apply at that node from a drop-down list. The tutoring system follows the *granularity* approach and requires the student to always select a node and then a transformation. If the student's selection is incorrect, the system will immediately display an error message and provide feedback. If the selections are correct, the system will apply the chosen transformation and update the trees - the student does not have to manually update the tree.

5 The Assessment Tool

Assessment is more than just determining whether an answer is correct or incorrect. A good assessment tool will also produce information about the type of error and the context to help the instructor or tutoring system determine how to best help a student resolve the error in her knowledge. The answers generated by a student when taking a test (pre or post) in our system are input to the grading tool which analyzes them and reports whether the student's answer(s) were correct or incorrect and additionally where the first error occurred, the type of error and the context in which the error was made. This information is used to help us build a model of the student's knowledge and the work on building and using the student model is reported in a companion paper [3].

There are three different types of possible errors: (i) incorrect node selection, (ii) incorrect transformation selection, and (iii) incorrect transformation application. Furthermore, some tree transformations can be applied in more than one context; for example, color flip at the root node is performed differently from color flip at a non-root node. The student might know how to correctly select and apply a transformation in one context but not in a different context. For each error, we would therefore also like to know the context in which it occurred.

The grading algorithm uses the constraints imposed on binary search trees (ordering and relationship of nodes) and its knowledge of red black trees to construct the canonical solution sequence. The key assumption is that the algorithm has a sufficient set of transformations (both primitive and those that are combinations of primitives) to recognize all transformations that a student might make. The algorithm assumes that a student will never combine more transformations than the system has. Thus the algorithm does not have to consider any intermediate steps that the student makes. If there is a step that the system does not recognize, then the step is skipped and the next step is checked to see if it is the result of some step or combination of steps. The result of the comparison steps is a sequence of transformations that the algorithm has identified. This sequence is compared to the sequence from the solution to try and find a match taking into account that a different ordering of transformations can also lead to a solution. This approach is sufficient to determine whether an answer is correct in all the cases that we have evaluated. If the answer is incorrect, the algorithm then proceeds (starting from the first step) to compare the answer and the solution step by step. Heuristics are used to modify the canonical solution to take into account macro-steps and reordered steps. These heuristics are sufficient to analyze most of the student answers that we have seen.

6 Data and Analysis

We evaluated the grading tool on three semesters of student data - Fall 2016, Spring 2017 and Fall 2017 - from a data structures class at our instituion. There was a total of 105 students from the three semesters.

The concepts underlying insertion were taught in lecture for one week and in the next week the students were evaluated over two days. On the first day, the students were given a pre-test for thirty minutes followed by a session with the tutoring system. Two days later, they were given the post-test which was a duplicate of the pre-test. The number of *unrecognized* errors is approximately

10% in the pre tests and 4% in the post tests. We analyzed the *unrecognized* errors by hand and found that in most cases we (the human graders) were also unable to determine the error type. Many of these errors were generated by students who were "gaming" the system to finish the tests faster by making random selections and then submitting the random answer. The data shows that the most common error was committed when selecting the appropriate transformation to apply. This error is caused by the student either (1) selecting the wrong node as the current node or (2) not correctly recognizing the preconditions for each transformation. The data provided in the student answers is insufficient for us to disambiguate between those two errors.

Table 1 shows a breakdown of the data by the type of transformation. The largest number of errors were made in applying the *color flip* transformation. This is one of the simpler transformations in that the colors of three nodes (current node, two children) are flipped. Most of the errors occurred due to the students not recognizing the applicable preconditions and selecting the transformation. Note that the grading tool only grades up until the first error so that if students were to make subsequent errors those errors would not be detected. This explains why the errors in *single rotation* and *double rotation* do not appear to decrease significantly (in some cases they increase) between the pre and post test. The *color flip* transformation frequently leads to a succeeding rotation and if a student does not apply the *color flip* they will not be tested on the second transformation. The tutoring helps the students recognize when to apply the *color flip* (decrease in errors between pre and post tests) and that leads them to an error in the subsequent rotation.

Table 1. Errors for each type of insertion operation

Semester	Color flip	Single rot	Double rot	Insertion	Recolor root	Unrecog
Pre F16	42	19	13	12	2	7
Post F16	15	16	17	6	1	2
Pre S17	67	25	19	30	3	13
Post S17	26	24	19	7	2	6
Pre F17	20	11	5	2	8	4
Post F17	12	6	9	3	1	4

Deletion operations are more complex than insertion operations and what makes it more difficult is that many of the operations share the same name as the insertion operations even though the preconditions and semantics are different. Just like for insertion, the most common error was committed when selecting the appropriate transformation to apply. The grading tool was able to correctly determine the type of error in approximately 90% of the pre and post test errors. Table 2 shows a breakdown of the data by the type of transformation. The largest number of errors were made in applying the *drop&rotate* transformation. This is a case that is poorly explained and illustrated in the textbook.

Table 2. Errors for each type of deletion operation

Semester	Color flip	Single rot	Double rot	Deletion	Recolor root	Switch value	Drop & rotate	Unrecog
Pre F16	1	15	19	4	6	11	50	9
Post F16	2	10	10	3	12	3	31	6
Pre S17	3	37	36	2	3	20	72	14
Post S17	1	18	22	2	10	5	63	13
Pre F17	0	14	14	0	1	19	35	11
Post F17	3	5	4	1	12	3	33	3

7 Conclusion

This paper has described an a tool for assessing student skills on red black tree (specialization of binary search tree) insertion and deletion algorithms. The advantage of the tool is that it provides a scaffolding-free (thus realistic) evaluation environment while still being able to accurately determine the correctness of student answers. In addition, it can classify 90% of the first error found in each student's answer. The tool uses strong domain knowledge and heuristics to determine the likely type of error in each case and has been evaluated on three semesters of student data from a data structures class. A companion paper shows how the analysis from the assessment tool can be used to construct an effective Bayesian student model.

References

1. Aho, A.V., Hopcroft, J.E., Ullman, J.D.: The Design and Analysis of Computer Algorithms. Addison-Wesley, Boston (1974)
2. Liew, C., Shapiro, J.A., Smith, D.: Determining the dimensions of variables in physics algebraic equations. Int. J. Artif. Intell. Tools **14**(1&2), 25–42 (2005)
3. Nguyen, H., Liew, C.W.: Building student models in a non-scaffolded testing environment. In: Proceedings of International Conference on Intelligent Tutoring Systems (2018)
4. Sedgewick, R.: Algorithms. Addison Wesley, Boston (1983)
5. Weiss, M.A.: Data Structures & Problem Solving Using Java, 3rd edn. Pearson Education Inc., London (2011)

Curriculum Pacing: A New Approach to Discover Instructional Practices in Classrooms

Nirmal Patel[1](✉), Aditya Sharma[1], Collin Sellman[2], and Derek Lomas[3]

[1] Playpower Labs, Gandhinagar, India
{nirmal,aditya.sharma}@playpowerlabs.com
[2] Arizona State University, Tempe, USA
collin.sellman@asu.edu
[3] Delft University of Technology, Delft, Netherlands
j.d.lomas@tudelft.nl

Abstract. This paper examines the use of "pacing plots" to represent variations in student learning sequences within a digital curriculum. Pacing plots are an intuitive and flexible data visualizations that have a potential for revealing the diversity of blended classroom instructional models. By using curriculum pacing plots, we identified several common implementation patterns in real-world classrooms. After analyzing two years' worth of data from over 150,000 students in a digital math curriculum, we found that a PCA and K-Means clustering approach was able to discover pedagogically relevant instructional practices.

Keywords: Curriculum analytics · Curriculum pacing
Sequence mining · Clustering · Visualization

1 Introduction

New data instrumentation methods have enabled digital learning products to capture large amounts of fine-grained data describing student behavior (e.g., clickstream data.) However, it is still a challenge to transform big educational data into real-world benefits for students and teachers. It is common for large-scale analyses to rely on simple, aggregated metrics like average score, total events, or total usage time. While these metrics are essential, they do not make use of the temporal dynamics of student actions present in the data and actualize the potential for big data in digital learning [9]. Our study was motivated by the potential to determine correlations between temporal trajectories of learners and various outcome measures such as student and teacher engagement, implementation fidelity, learning gains, fluency etc.

Different methods have been used to analyze educational sequence data, including association rule mining, sequential pattern mining, and process mining [10]. As we expand our capacity to analyze sequence data, certain challenges

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 345–351, 2018.
https://doi.org/10.1007/978-3-319-91464-0_38

arise. For instance, because of the diversity of usage behaviors, sequence mining techniques can produce overly complex and hard-to-interpret “spaghetti” models. Although these complex models can be used for predicting student actions over time more precisely, they have little interpretability. In this case, rather than analyzing sequence data directly, we can use clustering methods to group similar sequences together and analyze them separately [3,8].

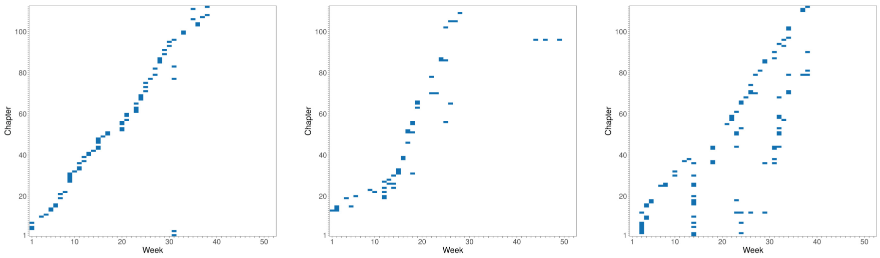
Sequence data from learners provide new opportunities to make data-driven intelligent tutors. For example, clickstream data have been used to produce next-step recommendations [7], induce reinforcement learning teaching strategies [12], and build data-driven pedagogical models [2]. These studies show that it is possible for intelligent tutors to base their recommendations upon models of sequential (or temporal) data.

2 Curriculum Pacing

One temporal aspect of classroom activity is what educators refer to as *pacing*, which has been described as “the rate at which new instructional material is introduced to students” [1]. This definition of pacing is, however, somewhat limiting, as it is common for students to revisit learning materials to ensure mastery before moving on to more advanced concepts. Thus, we define pacing as “the progression through curriculum over time.” This definition attempts to include all aspects of instruction over time in the construct of pacing. Although we have analyzed pacing in digital classrooms, we note that a great deal of classroom activity cannot be represented by the clickstream data.

It is straightforward to build a representation of pacing by using timestamped logs of student use of different curricular activities within a digital learning system. In addition to timestamps, the only other requirement is that learning activities should have attached metadata that indicate where the activity falls within a linear curriculum (e.g., unit 1, unit 2, unit 3, etc.) The notion of a linear curriculum remains inherent and central to the construct of pacing, which tells us how a student goes through the curriculum over time. However, even in cases where the curriculum is not entirely linear, pacing plots can still be used. For instance, each activity can be tagged with the average time when it is used in the curriculum. This approach can capture the linearity of activities that are used in sequence.

Our visual model of curriculum pacing aims to present student activity in the curriculum over time. This model has two dimensions: X dimension representing time (e.g., number of weeks) and Y dimension representing distance through the curriculum (e.g., unit 1, unit 2, etc.) Fig. 1 presents three examples of pacing plots. Pacing plots examples shown in Fig. 1 are just an instance of a broader visual design space. We explored various values for different design factors of the plot, including **Data** (single student, all students in a class,) **X-axis** (# of weeks, calendar data,) **Y-axis** (lesson number, average week used,) **Plot type** (scatter plot, heatmap,) and **Fills** (usage, score, percentile.) These variations were used to help identify different classroom implementation models across



(a) A pacing where the curriculum is covered in a linear fashion

(b) An accelerated pacing where the curriculum is covered quickly during middle of the year

(c) A pacing with "icicles" where the older content is revisited

Fig. 1. Examples of pacing plots. Each plot corresponds to a unique trace that one or more students left in the digital learning system. X-axis represents week of usage, and Y-axis represents the digital curriculum chapter numbers. Cells of the plot indicate whether student accessed the curriculum chapter in the given week or not.

a school district. The addition of score or percentile information can help to indicate where students or entire classrooms have struggled in a curriculum. We also found it useful to combine or average multiple plots to produce an aggregated pacing plot; e.g., to represent an entire district's "typical" pacing.

3 Meaningful Characteristic Variations

Classrooms have a great deal of diversity, and this variation is bound to appear in the pacing plots. During our initial review of pacing plots, we identified several characteristic variations in the form of the plots. These variations appeared to represent different classroom conditions and implementation approaches.

The most notable of all the patterns that appeared in these plots were vertical lines indicating that students are revisiting previous topics (e.g., "icicles") and horizontal lines indicating that students are practicing the same material over and over again (e.g., "ruts".) We identified several patterns that were likely to appear in classrooms: **Lockstep pacing**, a pattern where all students used the same material, represented as a tight pacing line; **Flexible pacing**, a pattern where there was variation in the material used, represented as a fuzzier pacing line; **Icicles**, a graph feature that appeared to occur when classes would engage in review; **Cram-to-complete**, the common tendency for an accelerated pace at the end of the school year; and **Glaciers** representing students entering at a later time during the year and catching up.

These variations were identified in a relatively small subset of data (< 100 classrooms.) To explore these variations at a larger scale, we surmised that a clustering method might be helpful for automating the identification and quantification of these pacing patterns. Effective clustering should be capable of revealing the patterns already identified and also, potentially, capable of revealing new patterns. We had two hypotheses for our study:

- **H1**: Clustering will identify previously known curriculum pacing variations that are meaningful to experts.
- **H2**: Clustering will identify previously unknown curriculum pacing variations that are meaningful to experts.

4 Participants and Method

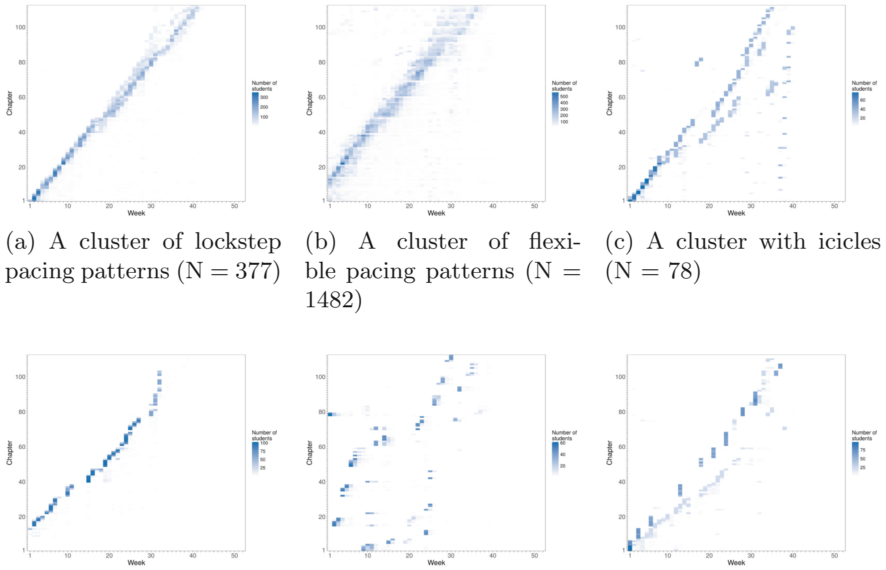
For our analysis, we used anonymized data from a large-scale online math curriculum that has been used by more than 150,000 students across the United States. The curriculum is divided into topics and subtopics; within each subtopic, there are a variety of activities such as videos, scaffolded practice quizzes, formative assessments, and homework assignments. The program is designed to go from the start to end in a linear fashion. Teachers assign resources to students, and students turn them in after completion. Using clickstream data from this program, we extracted pacing plots for individual students. Plots that were the same were merged. This deduplication reduced the size of the dataset by approximately 20% (N = 121,502). This produced a dataset with unique instructional patterns as data points, instead of students. This meant that we clustered different usage patterns, giving all the patterns same weight regardless of the difference in their frequencies.

To cluster pacing plots, we used K-Means clustering. Clustering high dimensional data can lead to the curse of dimensionality [5], where a large number of clusters is needed to discover meaningful patterns. Indeed, each of our pacing plots had 5824 features (52 weeks × 112 digital book chapters,) so we reduced the dimensionality of these data points using PCA as a preprocessing step. We used 212 principal components that explained 50% of the variance in data.

5 Results

We ran K-Means clustering on 121,502 unique instructional sequences using their lower dimensional representations. We chose a total of 50 clusters (K = 50,) which produced a model capturing 26.5% of the variation between the clusters. As the number of clusters was large, many clusters exhibited similar patterns, but this allowed us to find both unexpected and nuanced patterns. A large number of patterns did not have any characteristic variations in them. We found smaller clusters that identified both known and unknown characteristic variations in student learning sequences. We also found that many small clusters had little difference between them. Only a fraction of clusters are shown here.

In Fig. 2a, we see a group of instructional patterns where students accessed the same material with minimal variation (**Lockstep pacing**.) Fig. 2b shows a "fuzzier" pacing line, indicating that students who followed these patterns did different things in the same week (**Flexible pacing**.) Fig. 2c captures patterns



(a) A cluster of lockstep pacing patterns (N = 377)

(b) A cluster of flexible pacing patterns (N = 1482)

(c) A cluster with icicles (N = 78)

(d) A cluster with cram-to-complete pattern at the end (N = 101)

(e) A cluster where curriculum was followed in an unexpected manner (N = 66)

(f) A cluster where curriculum was accessed in bursts (N = 100)

Fig. 2. Visualizations of clusters from resulting analysis. Each plot is a combination of all data points in the cluster. Many of the patterns in the dataset had minimal usage, and meaningful clusters were smaller in size.

where, at the end of the year, previous chapters of the textbook were reviewed (**Icicles**,) and Fig. 2d shows patterns where materials were covered quickly at the end of the year (**Cram-to-complete**.) We did not find any clusters representing Glaciers, potentially due to our use of relative time in the pacing plots. Together, these clusters provide evidence that partially supports **H1**, that clustering will identify known curriculum pacing variations.

Figure 2e shows a group of unexpected instructional patterns: most of the curriculum was covered but in an unusual sequence. Some later topics were covered first, and some earlier topics were visited later. Figure 2f shows a group of patterns where the material was covered in bursts. These findings provide some evidence in the support of **H2**, that clustering will identify previously unknown pacing variations.

6 Discussion and Conclusion

Our approach shows how digital curriculum pacing plots might provide insight into variations in classroom instruction. Although we found evidence in support of our hypotheses, this preliminary work is limited. While clustering helped identify novel behavioral patterns, we have not evaluated their meaningfulness with, for instance, teachers who participated in those classrooms. We also suspect that the clustering techniques were unable to identify certain variations that were common in our classroom-level analyses. For instance, in our initial observations, we characterized different classrooms as "lockstep" or "personalized," based on the presence of "icicles"; these icicles appear to be cases where teachers were helping students by assigning them prerequisite skills. Why were these not observed in our clustering attempts? This may be because personalization behaviors were observed at a classroom level, whereas our analysis was focused on individual students. Alternatively, the dimensionality reduction approach that we took for clustering might be washing out finer details of the plots. This can be investigated in future analyses.

We expect that variations in pacing will predict variations in student outcomes [4,11]. In our future work, we will investigate the addition of student performance data in the pacing plots, which can illustrate nuances in instructional success at every step of the curriculum. These future approaches are intended to help understand and support classroom instructional practices at scale. One future possibility is that these models could support teacher-facing adaptive recommendation systems [6] that could help teachers learn from the aggregated decision-making of thousands of other teachers.

References

1. Barr, R., Dreeben, R.: How Schools Work, p. 33 (1991)
2. Chi, M., VanLehn, K., Litman, D.: Do micro-level tutorial decisions matter: applying reinforcement learning to induce pedagogical tutorial tactics. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6094, pp. 224–234. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13388-6_27
3. Hansen, C., Hansen, C., Hjuler, N., Alstrup, S., Lioma, C.: Sequence modelling for analysing student interaction with educational systems. In: Proceedings of the 10th International Conference on Educational Data Mining (2017)
4. Hoadley, U.: Time to learn: pacing and the external framing of teachers' work. J. Educ. Teach. Int. Res. Pedag. **29**(3), 265–277 (2003)
5. Keogh, E., Mueen, A.: Curse of Dimensionality. In: Sammut, C., Webb, G.I. (eds.) Encyclopedia of Machine Learning and Data Mining, pp. 314–315. Springer, Boston (2017). https://doi.org/10.1007/978-1-4899-7687-1_192
6. Koedinger, K.R., Brunskill, E., Baker, R.S., McLaughlin, E.A., Stamper, J.: New potentials for data-driven intelligent tutoring system development and optimization. AI Mag. **34**(3), 27–41 (2013)
7. Pardos, Z.A., Tang, S., Davis, D., Le, C.V.: Enabling real-time adaptivity in MOOCs with a personalized next-step recommendation framework. In: Proceedings of the 4th ACM Conference on Learning@Scale, pp. 23–32. ACM (2017)
8. Patel, N., Sellman, C., Lomas, D.: Mining frequent learning pathways from a large educational dataset. In: Proceedings of the 3rd International Workshop on Graph Educational Data Mining, pp. 27–30 (2017)
9. Reich, J.: Rebooting MOOC research. Science **347**(6217), 34–35 (2015)
10. Romero, C., Ventura, S., Pechenizkiy, M., Baker, R.S.: Handbook of Educational Data Mining. CRC Press, Boca Raton (2010)

11. Smith, J., Smith, B., Bryk, A.: Setting the pace: opportunities to learn in Chicago public elementary schools. Technical report (1998)
12. Zhou, G., Wang, J., Lynch, C.F., Chi, M.: Towards closing the loop: bridging machine-induced pedagogical policies to learning theories. In: Proceedings of the 10th International Conference on Educational Data Mining, pp. 112–119 (2017)

Towards Embedding a Tutoring Companion in the Eclipse Integrated Development Environment

Manohara Rao Penumala and Javier Gonzalez-Sanchez(✉)

Arizona State University, Tempe, USA
{mpenumal,javiergs}@asu.edu

Abstract. Programmers use Integrated Development Environments (IDEs) to write and test software, and students use them while learning programming. We explore the approach of embedding a tutoring companion inside Eclipse, a popular IDE. The embedded tutoring companion aims to be comparable to having an actual teaching assistant present all the time with each student throughout a course. The embedded tutoring companion tracks student's actions while solving a problem (coding, compiling, running) and collects metadata including the time spent, the correctness of the work, and the amount of copied or auto-generated code in the work. Then it can determine the practical understanding of the topics and concepts associated with the presented problem, it can assist the student by providing immediate feedback, and it can help instructors by reporting real-time information about students' performance. Our companion, implemented as an Eclipse plug-in, was evaluated with undergraduate students enrolled in a Java programming course.

Keywords: Companion · Teaching programming · Eclipse IDE

1 Introduction

Formative assessment (feedback for learning) helps the students to improve their understanding of the associated concepts. This statement holds true even more in the context of programming courses, as the student has to understand a concept properly in order to apply it or understand the successive concepts. Challenges in providing formative assessment by the instructors of a programming course include: (1) prompt assessment of a student's work and feedback delivery due to the limited amount of time per class session, besides the fact that instructors are not available 24/7; and (2) being aware of every student's individual performance due to the high number of students per class. These challenges can be addressed by automatizing the process of assessing students' work and then (1) automatizing the process of generating feedback and (2) keeping instructors informed of individual and group performances. A step towards this goal is presented here for a Java programming course. We propose an Eclipse-based tutoring companion that gathers data from students, provides feedback in real-time, and informs instructors about student and class performance. In the following sections, we present

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 352–358, 2018.
https://doi.org/10.1007/978-3-319-91464-0_39

the previous work in this domain (Sect. 2), define the architecture of our tutoring companion (Sect. 3), describe experiments conducted with the participation of undergraduate students, their results, and their implications (Sect. 4), and discuss the future work (Sect. 5).

2 Background

Using tutoring systems for complementing human teaching and as companions has been reported. For instance, Eitelman [1] discusses complementing human teaching with automated tutoring for teaching programming, examines reasons for the lack of extensive success of automated tutoring, and argues about the importance of a problem-based learning approach. In addition, Yang [2] describes the evolution and growth of intelligent tutoring systems as extracurricular assistants along with tutoring paradigms, student modeling, instruction modeling, adaptive curriculum planning, and user interfaces. Itkonen [3] reports preliminary experience using a test-driven lecturing approach to entwine assessment into course execution and argues that it allows early actions to improve learning. Ahankari and Jadhav [4] use e-rubrics as formative and summative assessment tools to evaluate the knowledge, understanding, and skill level of students. Fisher et al. [5] have explored the positive impact associated with the usage of online programming tools and have found that online programming tools have a quantifiable effect on the understanding and performance of students. Cain and Babar [6] discuss the usage of constructive alignment with formative feedback to help instructors gain better understanding of the level of students' knowledge in the course and to work towards improving that level.

The implementation of tutoring companions and systems for programming education has seen considerable success. Fernandes and Kumar [7] propose a tutor to teach the concept of static and dynamic scope in programming languages. The tutor evaluation shows considerable improvement in the overall retention of the concept by the students. Higgins et al. [8] report on the design, implementation, and usage of a course-based assessment system (CourseMarker). Students used it to solve programming exercises and submit their solutions. It is a heavy tool with multiple subsystems. Koh et al. [9] propose a cyber-learning tool that features so called real-time assessment of computational thinking. It enables educators to identify which concepts the students have mastered and which concepts the students are struggling with.

We propose a tutoring companion designed for Java based programming courses where Eclipse was already being used as the IDE. This solution, grounded in previous works, addresses two challenges: (1) making the use of the tutor natural and familiar by embedding it in the students' normal working environment, and (2) reducing installation lead-time. The solution implemented as a plug-in in the widely-used Eclipse IDE [10] does not ask the students to learn a completely new ecosystem in order to utilize the benefits of a tutoring companion. Moreover, the tutoring companion installation is the same as that for any other of the multiple plug-ins available for Eclipse IDE – a simple process guided step-by-step by in the IDE. Therefore, the students simply work on their tasks and activities without bothering about any additional steps they need to do because of the newly introduced tutoring companion.

3 Architecture

Our system architecture, depicted in Fig. 1, comprises three parts as follows:

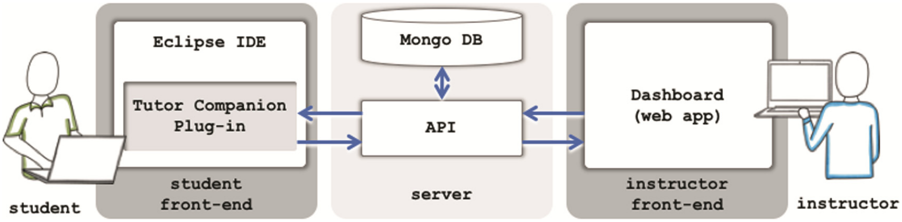
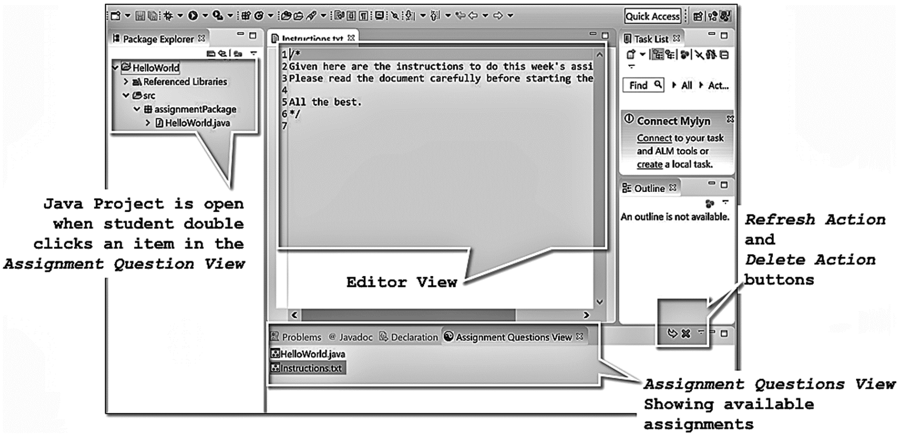


Fig. 1. Architecture including student front-end, server, and an instructor front-end.

Front-End for the Students. It implements the GUI for the tutoring companion *as a plug-in*. It shows assignments to the student and captures metadata associated with student's work. The plug-in will always prompt for a student ID and a course ID upon launch. Once student ID and course ID are entered, the student can move to the "Java perspective" (Fig. 2). In the Java Perspective, the student finds the "Assignment Questions View" at the bottom section of the window. Initially, there is no content in the view. The Assignment Questions View contains two buttons: "Refresh Action" and "Delete Action". A student clicks "Refresh Action" button to download active assignments, which are then listed in the Assignment Questions View. The student clicks "Delete Action" button to delete an assignment from the view and local machine. Double-clicking on any of the assignments listed in the Assignment Questions View



```
/*
Given here are the instructions to do this week's assi
Please read the document carefully before starting the

All the best.
*/

```

Connect to your task and ALM tools or create a local task.

An outline is not available.

Fig. 2. GUI for the student front-end running inside the Eclipse IDE.

opens a new Editor View that shows the instructions for the assignment and opens a new Java project in the Package Explorer View. While a student works on the Java project, editor actions, run and debug actions, and console messages are captured continuously and sent to the back-end server. The server will store them on the database and use them to infer feedback. After completing the assignment, a student clears the local data by deleting the project from disk and clicking on "Delete Action" button.

Back-End Server. It hosts a Web API connected to a MongoDB. Instructor accesses it to add and/or remove assignments for any number of courses. Assignments have a time constraint parameter to restrict the availability for download and timeframe for metadata capture. When the "Refresh Action" button is clicked on the Eclipse plug-in, assignments associated with the provided course and abiding to the time constraint are downloaded into the Assignment Questions View. The Eclipse plug-in reports the collected metadata to the API.

Front-End for the Instructor. The front-end, developed as a Web dashboard, allows instructors to monitor student and/or class performance per assignment and/or per topic. The dashboard is built on HTML5, CSS3, and JavaScript AJAX calls. The charts on the visualization view are drawn using D3.js and Google's Charts.js. The single page dashboard has two views: "Visualization View" and "Student View". The "Visualization View" interprets and represents the captured data across a class for a specific assignment question in five reports: student participation and success report, a pie chart depicting success and failure of the participating students in number; lines of code report, an area chart depicting the total lines of code in each participant's submission; number of run attempts report, a column chart depicting the number of times a student ran the program; submission time and compilation report, a line chart illustrating the time when the last metadata was captured; and runtime errors report, a bubble chart showing the compilation and runtime errors captured across all the students in a course for a specific question, which provides an insight into the areas where students are struggling. The "Student View" displays information about the course assignments that were downloaded by a student and provides detailed information about all the assignments completed.

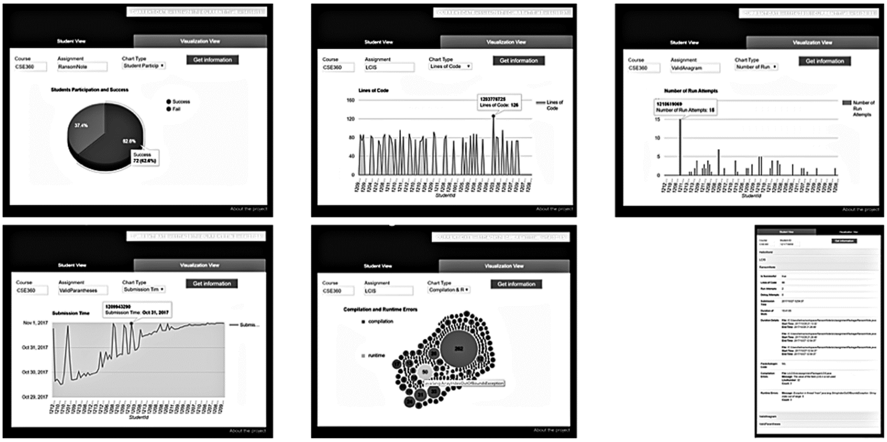
4 Usage and Discussion

Two experiments were conducted: a first one for identifying and fixing problems in the server and the tutoring companion interface, and a second one for gathering data.

Experiment 1. Twenty-five sophomore undergraduate students participated in this experiment. They were asked to complete four simple Java programming questions taken from LeetCode, a popular platform to practice coding. The assignments were Hello World, Degree of Array, Length of Last Word, and Valid Palindrome. Issues identified include: the plug-in is unstable on Eclipse Oxygen and on Eclipse Neon 3 running on Mac OSX; older versions of Java do not support the plug-in; and a bug allowed some students to access questions from wrong courses. Plug-in was updated and bug fixed but constrained to run on Java 8 and Eclipse Neon 3 recommended for Experiment 2.

Experiment 2. One hundred forty-five junior undergraduate students participated in experiment 2. They were asked to complete a Hello World program and four Java programming questions taken from LeetCode. The assignments were Hello World, Longest Subsequence, Ransom Note (Strings), Valid Anagram, and Valid Parentheses (Stacks). Participants were constrained to not print anything to console unless explicitly mentioned in the assignment.

Visualization View. The captured data supports a preliminary empirical assessment for the instructor front-end and possible inferences to be reported to the tutoring companion. For example, in the "student participation and success" report, the chart for the RansomNote assignment (shown in Fig. 3a) points the instructor to make a preliminary assumption about whether the participating students have understood the concept of Strings; in the "lines of code" report, the student ID highlighted in the chart shown in Fig. 3b has significantly more lines when compared to her/his peers – reviewing such code and helping in refactoring it can improve the student's approach towards the next assignments; in the "number of run attempts" report, the student ID highlighted in the chart shown in Fig. 3c has a curiously high number of run attempts, which implies poor understanding of the question or the associated concept – the student can be helped by providing more practice problems in the domain; in the "submission time" report, a chart that portrays too many plots at the end of the deadline could imply lack of clarity on the topic, a problem that is too lengthy or too complex, or student procrastination – the chart in Fig. 3d has a good mix of submission times, suggesting that the duration for the assignment work was sufficient; in the "compilation and runtime errors" report, shown in Fig. 3e, the ArrayIndexOutOfBoundsException was encountered 262 times, a considerable number – a drawback of the current implementation of the chart is that it displays insignificant errors, as well.


Course CSE360 | Assignment RansomNote | Chart Type Student Particip ▾

Course CSE360 | Assignment LCIS | Chart Type Lines of Code ▾

Course CSE360 | Assignment ValidAnagram | Chart Type Number of Run ▾

Course CSE360 | Assignment ValidParantheses | Chart Type Submission Tim ▾

Course CSE360 | Assignment LCIS | Chart Type Compilation & R ▾

Fig. 3. Top-down and left to right: (a) Student participation & success chart for RansomNote, (b) lines of code chat for LCIS, (c) number of run attempts chart for ValidAnagram, (d) submission time chart for ValidParantheses, (e) compilation & runtime errors chart for LCIS and (f) student view with details for a participant.

Student View. A full report of a particular student including all the assignments. For instance, Fig. 3f depicts the details of a RansomNote assignment question for a student who completed the question successfully, wrote 86 lines of code, ran the program 2 times, did not debug the code, submitted it a day before the deadline, spent more than 15 h on the IDE (not necessarily coding), did not copy/paste or auto-generate any code, and got the same compilation and runtime error 3 times each. Duration details list the time spent on all the files of the assignment question.

5 Conclusion and Future Work

The tutoring companion has the potential to become a dexterous tool in the learning process. Currently the tutoring companion is capable of tracking a student's actions and inferring the student's proficiency. There is no overhead on students to learn a completely new system since they have to do nothing more than installing a plug-in into an IDE that they already use. Future work in the short-term includes: (1) the improvement of the algorithm for establishing the legitimacy of the work done, for instance, by improving how to distinguish user-written code and copy/pasted code or auto-generated code; (2) tracking the keystrokes in the editor to clearly identify the time spent on coding vs. the time the editor was left idle; (3) generating lists of courses, assignments, and student IDs in the dashboard to provide ease of access; and (4) changing the run attempts count to better depict the student's confidence in her/his work. Long-term goals aim for a greater scope and scale including: (1) improving the check for correctness by adding checkpoints in assignment's life cycle; (2) building more visualizations that provide further insight into the work done; and (3) generating patterns from the data gathered across multiple semesters to build a coursework that enhances the learning further.

References

1. Eitelman, S.M.: Computer tutoring for programming education. In: Proceedings of 44th Annual Southeast Regional Conference, pp 607–610. ACM (2006)
2. Yang, F.J.: The ideology of intelligent tutoring systems. ACM Inroads **1**(4), 63–65 (2010). ACM
3. Itkonen, J.: Test-driven lecturing. In: Proceedings of 12th Koli Calling International Conference on Computing Education Research, pp. 141–142. ACM (2012)
4. Ahankari, S.S., Jadhav, A.A.: e-Rubrics: a formative as well as summative assessment tool for assessment of course and program outcomes. In: Proceedings of 8th International Conference on Technology for Education (T4E), pp. 246–247. IEEE (2016)
5. Fisher, W., Rader, C., Camp, T.: Online programming tutors or paper study guides? In: Proceedings of IEEE Frontiers in Education Conference (FIE), pp. 1–6. IEEE (2016)
6. Cain, A., Babar, M.A.: Reflections on applying constructive alignment with formative feedback for teaching introductory programming and software architecture. In: Proceedings Companion of 38th International Conference on Software Engineering (ICSE), pp. 336–345. IEEE/ACM (2016)
7. Fernandes, E., Kumar, A.N.: A tutor on scope for the programming languages course. ACM SIGCSE Bull. **36**(1), 90–93 (2004). ACM

8. Higgins, C.A., Gray, G., Symeonidis, P., Tsintsifas, A.: Automated assessment and experiences of teaching programming. ACM J. Educ. Resour. Comput. (JERIC) **5**(3), 3 (2005). ACM
9. Koh, K.H., Basawapatna, A., Nickerson, H., Repenning, A.: Real time assessment of computational thinking. In: IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC), pp. 49–52. IEEE (2014)
10. Murphy, G.C., Kersten, M., Findlater, L.: How are Java software developers using the eclipse IDE? IEEE Softw. **23**(4), 76–83 (2006). IEEE

Semantic Collaboration Trajectories in Communities of Practice

Matheus Pereira[1], Rosa Maria Vicari[1], and João Luis Tavares da Silva[2](✉)

[1] PPGC/UFRGS, Porto Alegre, Brazil
{mpereira,rosa}@inf.ufrgs.br
[2] UNIFTEC, Caxias do Sul, Brazil
joaoluis.tavares@gmail.com

Abstract. In communities of practice (CoP), learning occurs through constant interactions of their participants. The social aspect is fundamental for the construction of knowledge. This work uses semantic web technologies and ontologies to structure and represent the interactions of CoPs participants around a dynamic user profile. This user profile describes a set of dispersed properties and relationships in CoPs, allowing collaborative trajectories recovery in these learning environments.

Keywords: Communities of practice · Semantic web · Ontologies
User profile · Collaboration trajectory

1 Introduction

Communities of Practice (CoP) consist in groups of people who share a common interest and learn through continuous interactions [1]. The learner is an active agent that establishes relations, produces and socializes knowledge [2]. The social character of a CoP is fundamental to the knowledge construction process. It is through user interactions that bonds are created, experiences are shared, and the knowledge is explicited. For this reason, this work investigates how the dynamics of CoPs can be represented to describe collaboration trajectories in the context of learning, and try to answer the following question: is it possible to build a knowledge base capable of capturing the dynamic and distributed aspect of the interactions in communities of practice?

In order to answer this research question, we propose the use of semantic web technologies and ontologies to describe the relationships between the CoPs, their collaboration tools, contents and participants. The construction of this knowledge base will be explored to define a user profile that evolves while the participants interact and learn through regular exchanges. This dynamic profile allow us to represent collaboration trajectories, which map a group of properties and describe the forms of relationships that may occur in communities of practice, according to the 3C Collaboration Model [4].

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 359–365, 2018.
https://doi.org/10.1007/978-3-319-91464-0_40

2 Background

User profile is the process of managing and maintenance information associated with the user [5]. Studies involving information retrieval [6], content recommendation [7], adaptive virtual learning environments [5] and intelligent tutor systems [8] concentrate their efforts on this development. Knowledge, goals, interests, experiences and context are some of the information represented in user models. In the context of CoPs, the user interactions will be captured in order to follow, trace and analyze their learning path. The purpose of this approach is to identify the collaboration degree and the intensity of relations about CoPs participants in collaborative activities. The dynamic user profile consists in the semantic representation of the user interactions and involves the information sources relationship to their activities in the community. The capture and description of these actions will be used to represent the user collaboration trajectory in a given community.

Several researches use semantic web technologies to formalize user profiles [7,10,13], communities of practice [11,12] and collaboration in online communities [14,15]. The reuse of ontologies like FOAF (*Friend of a Friend*) and SIOC (*Semantically-Interlinked Online Communities*) also contribute to promote the information interoperability [13,15]. FOAF ontology [17] allows representing people and their social relationships. SIOC ontology [16] provides a vocabulary to represent online communities and user-generated content.

In this work we have applied semantic web technologies and an ontological representation, reusing FOAF and SIOC, in order to achieve a profile interoperability. This approach extends the information exchange possibilities and allows services sharing between applications.

3 A CoPPLA Ontology

In [2] a communities of practice framework is proposed with the objective of providing a semantic knowledge representation model for any CoP Platform (*CoPPLA*). In [3] a reference ontology was proposed in order to describe a general user profile in CoPs. This model focus on communities representation, its participants, interest profile and domain. The user profile has an identity, interactions, interests, roles and skills, classified in two levels: static and dynamic profile. An expanded CoPPLA ontology (Fig. 1) was conceived through studies on the real model of CoPs to represent the knowledge in a web CoP platform [18]. The relationships proposed are derived from actions that users can perform in the CoPs. The schema also includes FOAF and SIOC concepts.

In this ontology, the semantic structure of a community of practice (CommunityOfPractice) is a subclass of a community (sioc.Community). The community has a set of practices (has_practices, Practice) related to a domain of interest (related_domain, Domain). An online community (sioc.Community) has associated users (has_user, sioc.UserAccount) and a user is an extension of the semantic representation of SIOC online user. In addition, the user is associated with

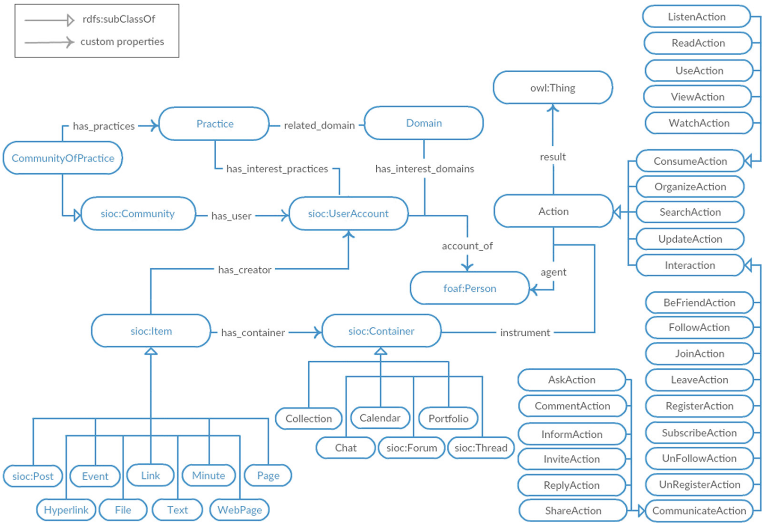


Fig. 1. CoPPLA ontology.

an FOAF representation (account_of, foaf.Person). Users (sioc.UserAccount) create content (has_creator, sioc.Item) in the CoPs' spaces for collaboration (sioc.Container). In these spaces the actions (Action) of the users (agent, foaf.Person) happen. These actions may be related to interaction (Interaction-Action), search (SearcAction), organization (OrganizeAction), update (Update-Action) or access (ConsumeAction) to the contents of a CoP. The user activity in a CoP is the result of an action that is part of a relationships set of the CoP-PLA platform. Therefore, the action class (Action) is fundamental to this work because it is from the user's actions that the dynamic profile, the interaction history and the collaboration trajectory are constructed.

4 User Profile and Collaboration Trajectory

The dynamic profile is the result of user actions on the platform. The semantic description of these actions increases the expressiveness and the ability to represent the information, allowing reasoning and inferences that may be explored to find complex relationships distributed in the environment. A simple information that can be retrieved from this representation is the user interaction history. This result can be obtained by means of an all-action query (Action) where the agent is an user in question. The representation ability and the dynamic aspect of the profile becomes apparent when it is possible to associate information to its history, such as the number of views of a particular content, the number of users that interacted in the same context, and related subjects and contents.

According to the 3C Model, in order to collaborate, individuals must exchange information (communication), operate together in a shared environment (cooperation) and organize themselves (coordination), assigning responsibilities and supervising each other [4]. Therefore, collaboration includes reciprocity and interdependency between pairs. Based on this definition, this work models an user collaboration trajectory as a historical set of communication actions among participants of CoPs in the same context. The availability of different tools for collaboration, as well as the ability to share and access content in CoPs, provide the necessary resources for communication and cooperation actions. Coordination occurs implicitly by organizing tools and CoPs structures and the commitments, conventions and vocabularies defined by the participants themselves during communication.

The identification of collaboration in communities of practice is done through a semantic query to relate users only in contexts where more than one participant has interacted by means of communication actions (CommunicateAction). In a semantic query that retrieves communication actions in the same context, the relations between the participants become apparent. This query highlights users who interacted with each other, while describing the path taken by the participant when navigating in the environment. In this way, the collaboration trajectory of a user is inferred from the relationships among participants on the same content, be it through sharing or discussions about a resource.

In this experiment we used the Communities of Practice Platform CoPPLA[1], that consists in a set of communication and collaboration tools for virtual CoPs instrumentalization. These tools involve the manipulation of texts, images, web pages, links, events, chats and spaces for learning experiences. Participants are able to create and manage their communities as a space to share knowledge involving learning activities [3].

This query definition allow us to follow the collaboration trajectory from the perspective of individual users, but also, the association with the other collaborative activities in the various contexts of a CoP, representing the collective production of the participants and the community of practice, displaying its practice and domain.

For a basic example, using CoPPla ontology our semantic server environment may serve a query like: "SELECT ... WHERE ?user a foaf:Person . ?user foaf:topic_interest ?topic . ... ?action coppla:context ?resource . ?type rdfs: sub-ClassOf* coppla:Action . ... FILTER(?user = <URL.../coppla/author/ John>)...", in order to get John's collaboration trajectory as illustrated in Table 1.

5 Results and Discussion

This work propose a mapping of relationships among CoPs, collaboration tools, participants and their interactions, proposing a semantic representation for the dynamically constructed knowledge in CoPs. The proposed solution establishes services for the acquisition, persistence and recovery of interactions in CoPs.

[1] http://www.coppla.com.br/.

Table 1. Example of John's Collaboration Trajectory.

context_title	*user1*	*type action1*	*user2*	*type action2*
Collaborative filtering	John	Share	Rosa	Comment action
Collaborative filtering	John	Reply	Rosa	Comment action
FOAF/SIOC ontologies	John	Reply	Matheus	Comment action
FOAF/SIOC ontologies	John	Reply	Rosa	Reply action
FOAF/SIOC ontologies	John	Reply	Matheus	Share action
OBAA pattern	John	Comment	Clara	Comment action
OBAA pattern	John	Comment	Clara	Share action
OBAA pattern	John	Comment	Jose	Reply action

A semantic server stores the RDF triples described with the CoPPLA, FOAF and SIOC ontologies, allowing the execution of semantic queries and inferences. These queries retrieve information from the users and their dispersed interactions in the various collaboration tools of a CoP. This information can be used to find interests, historical interactions, and collaboration trajectories in CoPs.

The first contribution of this work was the adequacy of ontologies and the use of semantic web technologies to formalize the environment information. From the CoPs knowledge formalization, the information interoperability was improved and semantic queries and automatic processing became possible to be performed. The user profile with associated semantics is able to store information that was previously scattered among different CoP collaboration tools. The ability to track user interactions, capturing different aspects of their interactions, and representing them with semantic value allows to combine, reuse, and share the knowledge dynamically constructed during exchanges between participants.

Semantic queries have the ability to retrieve information related to the static and dynamic aspects of the participants. The proposed representation is capable of organize information that is linked to both the user and their relationship network. Thus, the dynamic profile is updated according to the user's actions and with their colleagues actions. To these actions it is possible to associate the context and the moment in which they occurred, the participants involved, the type of action executed and the collaboration tools used. These information, organized from an individual perspective, allows the retrieval of interaction histories and collaboration trajectories. This may be explored to understand how knowledge is built on CoPs and allows the construction of new collaboration tools based on the behavior pattern of each participant.

Future work intends to evolve the user profile and aspects related to the performance, security and privacy of semantic queries. The use of the semantic web also allows the execution of federated queries that can access resource descriptions on external semantic bases. Thereby, it is possible to search for new relationships and combine information to generate new knowledge. Finally, the organization of the user actions in a formal representation allows the execution of complex queries, retrieving and combining information, including incomplete ones, to discover new knowledge. From this information, it is possible to create interactive dashboards combining user actions, contexts, other participants who interacted in the same resource, related materials and related topics. The dynamic user profile evolves as the interactions occur in the CoPs and may be explored to adapt the platform and to improve recommendation systems.

References

1. Wenger, E.: Communities of practice: learning as a social system. Syst. Think. **9**(5), 23 (1998)
2. Ribeiro, A.M., Silva, J.L., Boff, E., Viccari, R.M.: Dos ambientes de aprendizagem às comunidades de prática. Simpósio Brasileiro de Informática na Educação, vol. 22 (2011)
3. Da Silva, J.L., Ribeiro, A.M., Boff, E., Primo, T.T., Viccari, R.M.: A reference profile ontology for communities of practice. Int. J. Metadata Semant. Ontol. **7**(3), 185–196 (2012)
4. Fuks, H., Raposo, A.B., Gerosa, M.A., Lucena, C.J.: Applying the 3C model to groupware development. Int. J. Coop. Inf. Syst. **14**(02n03), 299–328 (2005)
5. Brusilovsky, P., Millán, E.: User models for adaptive hypermedia and adaptive educational systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) The Adaptive Web. LNCS, vol. 4321, pp. 3–53. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72079-9_1
6. Ghorab, M.R., Zhou, D., Oconnor, A., Wade, V.: Personalised information retrieval survey and classification. User Model. User-Adap. Inter. **23**(4), 381–443 (2013)
7. Primo, T.T., Vicari, R.M., Bernardi, K.S.: User profiles and learning objects as ontology individuals to allow reasoning and interoperability in recommender systems. In: Global Engineering Education Conference (EDUCON), pp. 1–9. IEEE (2012)
8. Käser, T., Klingler, S., Schwing, A.G., Gross, M.: Beyond knowledge tracing: modeling skill topologies with bayesian networks. In: Trausan-Matu, S., Boyer, K.E., Crosby, M., Panourgia, K. (eds.) ITS 2014. LNCS, vol. 8474, pp. 188–198. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-07221-0_23
9. Fenza, G., Orciuoli, F.: Building pedagogical models by formal concept analysis. In: Micarelli, A., Stamper, J., Panourgia, K. (eds.) ITS 2016. LNCS, vol. 9684, pp. 144–153. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39583-8_14
10. Plumbaum, T.: User modeling in the social semantic web (2015)
11. Tifous, A., El Ghali, A., Dieng-Kuntz, R., Giboin, A., Christina, C., Vidou, G.: An ontology for supporting communities of practice. In: Proceedings of the 4th International Conference on Knowledge Capture, pp. 39–46. ACM (2007)
12. Chikh, A., Berkani, L.: Communities of practice of e-learning, an innovative learning space for e-learning actors. Procedia-Soc. Behav. Sci. **2**, 5022–5027 (2010)
13. Fernandez, M., Scharl, A., Bontcheva, K., Alani, H.: User profile modelling in online communities (2014)
14. Caballé, S., Daradoumis, T., Xhafa, F., Juan, A.: Providing effective feedback, monitoring and evaluation to on-line collaborative learning discussions. Comput. Hum. Behav. **27**(4), 1372–1381 (2011)
15. Conesa, J., Caballé, S., Gañán, D., Prieto, J.: Exploiting the semantic web to represent information from on-line collaborative learning. Int. J. Comput. Intell. Syst. **5**(4), 653–667 (2012)

16. Bojars, U., Breslin, J.G., Berrueta, D., Brickley, D., Decker, S., Fern andez, S., Gorn, C., Harth, A., Heath, T., et al.: SIOC core ontology specification (2007)
17. Brickley, D., Miller, L.: FOAF Vocabulary Specification (2012)
18. Da Silva, J.L.T., Ribeiro, A.M., Reategui, E.: Tecnologias semânticas aplicadas a representação de conhecimento educacional em comunidades de prática (2016)

Predictors and Outcomes of Gaming in an Intelligent Tutoring System

Chad Peters[1], Ivon Arroyo[2], Winslow Burleson[3], Beverly Woolf[4], and Kasia Muldner[1](✉)

[1] Institute of Cognitive Science, Carleton University, Ottawa, Canada
kasia.muldner@carleton.ca
[2] Social Science and Policy Studies, WPI, Worcester, USA
[3] Rory Meyers College of Nursing, NYU, New York, USA
[4] College of Information and Computer Sciences, UMass, Amherst, USA

Abstract. In the present paper we present analysis of gaming actions with MathSpring, an established ITS for mathematics for high school students. Our findings indicate that both student and problem features were similarly predictive of gaming behaviors, as well as that gaming was associated with lower excitement and lower learning gains.

Keywords: Gaming · Predictors · Outcomes · Affect
Intelligent tutoring system

1 Introduction

A student is said to be gaming an intelligent tutoring system (ITS) when "*they attempt to succeed in an educational task by systematically taking advantage of properties and regularities in the system used to complete that task, rather than by thinking through the material*" [1]. Since the seminal gaming work by Baker et al. [1], there have been various studies exploring the causes of gaming and/or ways to detect it (e.g., [2, 3]), as well as ones investigating the impact of gaming on learning [4–6]. In this paper, we focus on whether it is the student or the ITS design that drive gaming behaviors, but also do explore the impact of gaming on learning.

What is a stronger predictor of gaming, student traits or ITS features? On the one hand, research on individual differences suggests that student traits should be highly predictive of gaming (given that the definition of gaming indicates this behavior is the result of a student trying to avoid thinking). For instance, seminal work on self explanation indicates that while some students spontaneously think constructively about instructional material (e.g., by making inferences over and beyond the presented text), others resort to shallow reading strategies like skimming text that avoids thinking too deeply about the material [7]. As another example, studies involving tutoring systems demonstrate that under some conditions, students prefer to copy solutions from examples rather than generate the answer themselves [8]. Beyond student characteristics, however, the learning context itself, such as a tutoring system or classroom activity, also has an impact on how students interact with it [2, 9]. For instance, if a

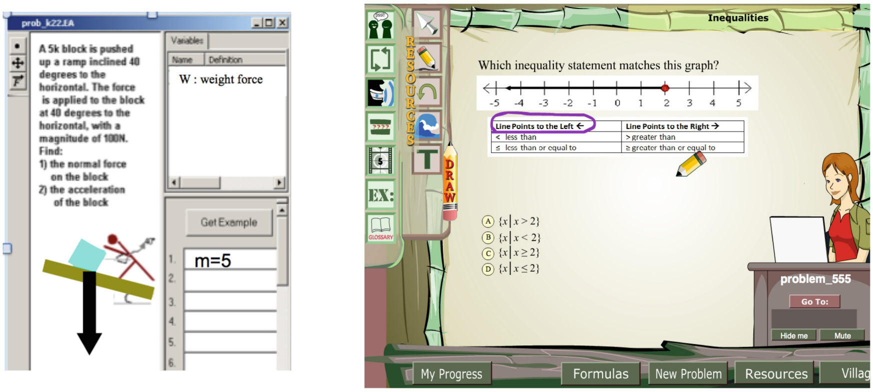
R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 366–372, 2018.
https://doi.org/10.1007/978-3-319-91464-0_41

tutoring system facilitates shallow behaviors through poor design of hints or problems, then students will abuse the corresponding functionalities by gaming [2].

Prior work within the specific context of ITSs has examined the question of what drives gaming behaviors, student traits vs. ITS features, by labeling student actions within an ITS as gamed or not, and subsequently analyzing what type of feature best predicts gaming actions [2, 9, 10]. The findings have been mixed. Some analyses have shown that gaming is best predicted by the features of the ITS (as opposed to the student). For instance, the model in [2] that used a set of tutor-related features explained 56% of the variance in gaming, higher than previous attempts to explain gaming. In another study, Baker [11] directly compared student vs. ITS features as gaming predictors and found that the latter were a better predictor. Recently, research in [12] suggested that tutor features had a larger impact than student features on how students game the system (i.e., patterns of gaming behaviors) [12]. In contrast, Muldner et al. [10] found that the student working on a given problem was almost twice as strong a predictor of gaming as compared to the problem being worked on. This data came from a study involving a physics ITS called Andes used by college students. However, they also found in a secondary analysis with the Cognitive Tutor ITS and middle school students that both student and ITS features were similarly strong predictors of gaming. In general, these variations in findings point to the need for more research.

Other work focused on student emotion as a potential predictor of gaming. For instance, Baker et al. [13] found that boredom was the primary predictor of gaming, which in turn diminished learning gains.

In the present analysis, we revisit the question of gaming predictors and outcomes. We follow the methodology in [10] but apply it to a different ITS and population. Muldner et al.'s [10] primary analysis involved data from the Andes ITS. Andes, shown in Fig. 1 (left) made gaming of some solution entries difficult, since they corresponded to physics equations that students free-typed into the interface (and thus systematically guessing to arrive at the solution was not productive). However, Andes did provide gaming opportunities through hints (abstract features of hints is one of the



A 5k block is pushed up a ramp inclined 40 degrees to the horizontal. The force is applied to the block at 40 degrees to the horizontal, with a magnitude of 100N.
Find:
1) the normal force on the block
2) the acceleration of the block

W : weight force

1. m=5
2.
3.
4.
5.
6.

Which inequality statement matches this graph?

-5 -4 -3 -2 -1 0 1 2 3 4 5

Line Points to the Left ←	Line Points to the Right →
< less than	> greater than
≤ less than or equal to	≥ greater than or equal to

Ⓐ $\{x \mid x > 2\}$
Ⓑ $\{x \mid x < 2\}$
Ⓒ $\{x \mid x \geq 2\}$
Ⓓ $\{x \mid x \leq 2\}$

Fig. 1. The Andes ITS (left) and the MathSpring ITS (right)

features of ITS design that promote gaming [2]). In the present analysis, the data came from a different ITS, namely MathSpring, and a younger population (middle school students) rather than the college population in [10]. The following research questions guided our analysis:

(1) What is a stronger predictor of gaming (the student or the ITS features)?
(2) Is gaming associated with reduced excitement or interest?
(3) Is gaming associated with diminished student learning?

2 The Data and the Gaming Detector

The data for the present analysis comes from a prior study with grade seven students (N = 191) using MathSpring [14]. The study was conducted within the students' school and spanned three consecutive school days – details are in [14]. All problems in MathSpring used the multiple-choice format with four options for each question; MathSpring provided immediate feedback by coloring an entry red (incorrect) or green (correct), at which point the student moved on to the next problem. MathSpring also made hints available, accessed by clicking on the hint button. To obtain information on how students were feeling as they solved problems, MathSpring prompted students to self-report their affect at regular intervals on two target emotions: interest and excitement (see [14] for details).

Given that the primary goal of the present analysis was to compare the predictors of gaming with results in [10], we followed that prior methodology for the construction of the gaming detector. Thus, the gaming detector was based on analyzing the time between a series of *tutor-student* action pairs in the target ITS, where (1) *tutor actions* corresponded to providing feedback by coloring a student entry red or to providing a hint and (2) *student actions* corresponded to either generating a new solution attempt, or asking for a hint. Given this framework, we labeled the following three primary tutor-student pairs as **gamed** for the present analysis: (1) *Guessing:* the tutor signals an incorrect entry, and the student quickly generates another entry (see 'G – incorrect' and 'G – correct' cells in Table 1); (2) *Skipping a hint to ask for another hint:* the tutor presents a hint and the student skips the hint by quickly asking for another hint (see '*S1'* cell in Table 1); (3) *Skipping a hint in the context of solution generation:* the tutor presents a hint and the student quickly generates a solution entry (see '*S2'* cell in Table 1). This action could be indicative of a student ignoring the hint to guess instead at the answer.

An important aspect of gaming detection pertains to the time threshold used to identify whether an action was gamed - we used the method in [10] to determine this threshold by visually inspecting the time distribution of each tutor-student pair under consideration. The thresholds used to distinguish gamed pairs from not gamed pairs corresponded to ones used by the Andes detector, because the distributions for the present data indicated they were appropriate (skipping hints in the context of a hint series (S1) < 3, guessing < 4 s; and skipping hints in the context of solution generation (S2) < 4 s).

Table 1. Gaming opportunities for *Tutor–Student* pairs (cells corresponding to gaming are shaded). Each cells shows the mean % of responses over all the students (note that this is not per student), and in parentheses the mean % of a student response for that row's tutor action.

	(a) Student: Hint Request		(b) Student: Entry		
	fast	slow	fast		slow
Tutor: Hint	S1: 11 (*28*)%	19 (*46*)%	S2: .4 (*1*)%		10 (*25.3*)%
Tutor: Incorrect	.2 (*0.4*)%	1.5 (*2.5*)	G - correct 15 (*26*)%	G - incorrect 10 (*16.7*)%	32(*54.3*)%

Legend: *fast: student action < gaming threshold; slow: student action >gaming threshold*

3 Results

We begin with the descriptive data. Following the approach [10], we used *problem* as the unit of analysis, and calculated the percentage of gamed *tutor-student* action pairs on a given problem (this is preferable to using raw values, since it normalizes the gaming data making it comparable across students and problems). The overall mean percentage of gaming per student across all problems was 13.8%. Table 1 shows the distribution of gaming behaviors (collapsed across students to make it comparable to [10]). *Guessing* leading to correct solutions is quite a bit higher than it was in the Andes data (see the *G – correct*) - 15% in the present analysis vs. 5% in the Andes data. This is not surprising given that the MathSpring uses a multiple choice format, making this type of guessing feasible in MathSpring than in Andes.

What is a Better Predictor of Gaming, Student or Problem (RQ1)? To see if student or problem features better predict gaming, we followed the approach in [10] by obtaining measures on gaming as follows:

(1)	PercentageGaming$_{s\ p}$	Percentage of gaming by a student s on a problem p
(2)	$\sum_{p=0}^{p=N} \text{PerGaming}_{sp}/N$	Average gaming by a student s across all problems p solved by that student
(3)	$\sum_{s=0}^{s=N} \text{PerGaming}_{sp}/M$	Average gaming on a problem p across all students s

We then conducted a linear regression with *percentageGaming* as the outcome variable (Eq. 1) and as its two predictors, the average gaming by a student across all problems (Eq. 2, referred to as *student* below) and the average gaming on a problem across all students (Eq. 3, referred to as *problem* below). The resulting model explained

17.7% of the variance ($R^2 = 17.7$)[1]. In this model, both *student* and *problem* are significant predictors of gaming ($p < .01$) but their standardized coefficients are close in magnitude: *student* = .309, vs. *problem* = .271). This suggests that both *student* and *problem* are comparable predictors of gaming in MathSpring, in contrast to what was found in [10] for the Andes data, where *student* was almost twice as strong a predictor as *problem*.

Is Gaming Associated with Reduced Excitement or Interest (RQ2)? To answer RQ2, we conducted two partial correlations between gaming (formula 2 above) and each of the two emotion variables (excitement, interest), after controlling for pre-test (this was done to control for a priori knowledge, since this could be another factor influencing gaming frequency). We found that gaming was negatively associated with excitement, $r(129) = -.20$, $p = .024$, with students who gamed more self-reporting lower excitement. The same pattern emerged for the interest variable, although this trend did not reach significance, $r(136) = -.13$, $p = .14$.

Is Gaming Associated with Diminished Student Learning (RQ3)? To answer our third research question we correlated gaming frequency (formula 2 above) with learning gains (post – pre). Gaming was negatively associated with learning, r $(139) = -.25$, $p = .002$, with students who gamed more obtaining lower pre to post gains.

4 Conclusion and Future Work

In the present analysis, we build a gaming detector and used its output to show that (1) both students and the problems they solved were comparable predictors of gaming, (2) gaming was associated with lower excitement and interest, and (3) gaming was associated with less learning (as measured by pre to post test gains).

Here, we focus our discussion on result (1), since this corresponds to our primary analysis. In prior work [10], the *student* solving a given problem in the Andes ITS was a stronger predictor of whether the problem would be gamed than the *problem* being solved. This opened up the possibility student characteristics are a stronger predictor of gaming. In contrast, in the present analysis we found that both the *student* and the *problem* the student was solving were comparable predictors whether a tutor-student pair would be gamed. The present result may be a function of the ITS, in that MathSpring offers multiple choice questions that may be tempting for students to abuse by guessing instead of by deriving the solution on their own, which in turn could explain the discrepancy between the current and prior results. However, students in the present analysis gamed less overall than in the Andes data set. This suggests that the population using the ITS has a role in what influences gaming: the present analysis involved students in middle school, which was also the target population in other gaming studies that found ITS features to be strong gaming predictors [2, 9], as compared to college

[1] This analysis uses dependent samples, which impacts the validity of the p value. While $p < .001$ in our analysis, this is not our focus – we are interested in the model parameters, and these will still be valid under the conditions of the present analysis (e.g., A. Field, 8.3.2.1, *Discovering Statistics*).

students (the target population for the Andes detector). Thus, it may be that even though *overall* younger populations tend to game less than older populations, when the former do game, they are more swayed by ITS features, for instance because they have not yet developed their mega-cognitive skills needed to engage in effective learning behaviors.

The gaming detector used in the present study was based on a prior methodology [10] because we wanted to compare our results to its results. This detector focuses on tutor-student action pairs, such as *tutor provides hint – student skips* hint. The fact that such quick action pairs should be considered gaming is not controversial, but more complex patterns of patterns of gaming have been proposed by human coders in other research (e.g., [9]). These action patterns go over and beyond the essentially baseline analysis provided by the detector used in the present study, and so an interesting avenue of future work involves seeing how the current results apply using human coders of gaming.

Acknowledgements. This research was supported by NSERC discovery grant #1507 and NSF grant #1324385.

References

1. 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 (2004). https://doi.org/10.1007/978-3-540-30139-4_50
2. Baker, R., et al.: Educational software features that encourage and discourage "gaming the system". In: Proceedings of AIED, pp. 475–482 (2009)
3. Walonoski, J.A., Heffernan, N.T.: Detection and analysis of off-task gaming behavior in intelligent tutoring systems. In: Ikeda, M., Ashley, K.D., Chan, T.-W. (eds.) ITS 2006. LNCS, vol. 4053, pp. 382–391. Springer, Heidelberg (2006). https://doi.org/10.1007/11774303_38
4. Baker, R., et al.: Off-task behavior in the cognitive tutor classroom: when students "game the system". In: Proceedings of CHI 2004, pp. 383–390 (2004)
5. Aleven, V., et al.: Toward meta-cognitive tutoring: a model of help-seeking with a cognitive tutor. Int. J. Artif. Intell. Educ. (IJAIED) **16**(2), 101–128 (2006)
6. d Baker, R.S.J., Mitrović, A., Mathews, M.: Detecting gaming the system in constraint-based tutors. In: De Bra, P., Kobsa, A., Chin, D. (eds.) UMAP 2010. LNCS, vol. 6075, pp. 267–278. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13470-8_25
7. Chi, M.T.H., et al.: Self-explanations: how students study and use examples in learning to solve problems. Cogn. Sci. **13**(2), 145–182 (1989)
8. Muldner, K., Conati, C.: Scaffolding meta-cognitive skills for effective analogical problem solving via tailored example selection. Int. J. Artif. Intell. Educ. **20**(2), 99–136 (2010)
9. Hawkins, W., Baker, R., Heffernan, N.: Which is more responsible for boredom in intelligent tutoring systems: students (trait) or problems (state)? In: Affective Computing and Intelligent Interaction (2013)
10. Muldner, K., et al.: An analysis of students' gaming behaviors in an intelligent tutoring system: predictors and impacts, user modeling and user adapted interaction. J. Personal. Res. Spec. Issue Data Mining Personal. Educ. Syst. **21**, 99–135 (2011)

11. Baker, R.: Is gaming the system state-or-trait? Educational data mining through the multi-contextual application of a validated behavioral model. In: Workshop on Data Mining for User Modeling, pp. 76–80 (2007)
12. Paquette, L., Baker, R.S.: Variations of gaming behaviors across populations of students and across learning environments. In: André, E., Baker, R., Hu, X., Rodrigo, M.M.T., du Boulay, B. (eds.) AIED 2017. LNCS (LNAI), vol. 10331, pp. 274–286. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-61425-0_23
13. Baker, R., et al.: 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. Int. J. Hum. Comput. Stud. **68**(4), 223–241 (2010)
14. Muldner, K., Wixon, M., Rai, D., Burleson, W., Woolf, B., Arroyo, I.: Exploring the impact of a learning dashboard on student affect. In: Conati, C., Heffernan, N., Mitrovic, A., Verdejo, M.F. (eds.) AIED 2015. LNCS (LNAI), vol. 9112, pp. 307–317. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-19773-9_31

Classifying Interaction Behaviors of Students and Conversational Agents Through Dialog Analysis

Michael Procter[1], Robert Heller[2], and Fuhua Lin[1](✉)

[1] School of Computing and Information Systems, Athabasca University, Athabasca, Canada
{mprocter,oscarl}@athabascau.ca

[2] Faculty of Humanities and Social Sciences, Athabasca University, Athabasca, Canada
bobh@athabascau.ca

Abstract. E-learning systems based on a conversational agent (CA) provide the basis of an intuitive and engaging interface for the student. The goal of this paper is to propose a method for detecting conversational interaction behaviors of learners and CAs, using an agent-based framework, for the purpose of improving the communication between students and CA-based intelligent tutoring systems. Our framework models both the student and the CA and uses agents to represent data sources for each. We show how the framework uses the detection of conversational behaviors to initiate interventions to improve student conversational engagement. The results of initial user testing are reported.

1 Introduction

Conversational agents (CAs) provide users with the ability to interact with computer software using natural language. CAs embedded within e-learning applications have potential to provide an intuitive, user-friendly interface that engages students [1].

We seek to improve the interaction between students and e-learning CAs through maintaining student engagement. Evaluating user engagement typically involves devices for sensing verbal and non-verbal behavior cues [2]. Eye trackers [3], cameras [4], and EEG headsets have been employed but non-intrusive collection of data is required to make engagement-aware applications a practical reality [5].

One of the least intrusive approaches to evaluate engagement is the analysis of the conversation between the user and the CA [6]. This has not been explored extensively in the literature, despite the substantial amount of research associated with text-based affect detection. [7] describes a technique for measuring cognitive engagement based on Turney's level of word abstraction dictionary to distinguish between forum posts which are more descriptive and those that are more interpretive [8]. Our research breaks new ground by detecting users' engagement based on a real-time analysis of contributions made to the conversation by the user.

Specifically, we propose a method for improving student conversational engagement by identifying student conversational behaviors through analysis of the conversational

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 373–379, 2018.
https://doi.org/10.1007/978-3-319-91464-0_42

log, using text-analysis techniques. We derive appropriate interventions for delivery by a CA, and test the method using online psychology students.

2 Student Conversational Behavior Identification

For a CA-based intelligent tutoring system (ITS) focusing on the conversational interface, it is important to measure if conversation is being used effectively for the learning task. As with human-human conversation, when students converse with a CA-based ITS it is important to evaluate if they are engaged in: the "act" of conversing; the topic or task associated with the conversation; and creating responses relevant to the CA's comments. Ideally, the software should be able to monitor these factors and make a real-time assessment of "how the conversation is going".

We describe a method of judging the quality of the students' conversational responses with a CA-based ITS, and demonstrate this on a CA that simulates chatting with an historical figure. We generated algorithms to categorize student input by identifying conversational behavior patterns. Identifying problematic conversational behaviors allows for targeted interventions to repair or improve the conversation.

2.1 Algorithms

Algorithms were based on data from previous research using a CA simulating an historical figure, Sigmund Freud [9]. To learn more about Freud, students chat with Freudbot using text input, as if interviewing him. Freudbot responds in first person to questions and comments about Freud's life, family, theories, and colleagues, and follows rules of conversation, such as turn-taking, and repairing misunderstandings.

Conversational logs of previous interactions with the CA were examined for common patterns of user behavior. Behaviors of interest were: (1) robust, repeated patterns of interaction; (2) detectable with data available to the CA (e.g. type of output from the CA, user word count); (3) indicated how well the interaction was proceeding.

Three behaviors were identified:

Tryer: The user attempts to ask questions using reasonably full sentences on Freud-related topics. They continue to do this despite little or no success in getting Freud-related information from the CA. This *trying* behavior is characterized by relatively long sentences, high number of no-match cases per inputs and possibly input words with high abstractness value, a measure of cognitive engagement [7].

Keyworder**:** The user answers questions or responds to CA output with single words or phrases associated with Freud or psychoanalysis. E.g. "ego", "psychoanalysis", "anxiety", typically jumping from topic to topic. This *keywording* behavior is detected by low word count, non-repetition, and low number of no-match cases per inputs.

***Morer*:** The user discovers a word that leads to advancement through the narrative (e.g. “ok”), and repeats that word. *Moreing* behavior can be detected by recognizing back-channel type words and phrases (“more”, “ok”, “I see”), and high consecutive repetition of those words.

All three of the behaviors require recognizing a repeated pattern and therefore evaluate over a period of time. For each user response to CA output, the CA text is categorized into one of 15 output types (e.g. yes/no question, domain content). User text is checked for word count, ‘more’ type words, and repetition of the last response. Occurrences of behaviours are counted based on these measures. If a behavior count exceeds a threshold for that behavior type, the user is categorized accordingly. Thresholds were tuned to reduce false positives with minimal false negatives. False positives are likely to trigger inappropriate interventions. This is confusing to the user, and undermines the perception of intelligence that plays a large part in student engagement.

To evaluate and optimize the algorithms, we manually rated 26 conversations (613 turn pairs) from the chat logs of a previous experiment [9]. Each conversation was assigned a rating for each type of behavior: *trying*, *keywording*, and *moreing*. Results from comparing the manual and automated ratings are shown in Table 1.

Table 1. Behavior algorithm testing

	True positive	True negative	False positive	False negative	Total
Tryer	12	7	1	6	26
KW	3	17	6	0	26
More	2	23	0	1	26

2.2 Algorithm Evaluation

Table 1 provides numbers that were useful for tuning the algorithms and selecting the best parameters. To verify the accuracy of the algorithms, chat logs from the current study (see Sect. 3) were manually coded to identify the three behaviors. The human coder read the entire log for a participant and assigned any behaviors observed, and a confidence rating from 1 (low) to 3 (high) for each behavior.

The agent’s behavior assignments were compared against those of the human coder. Observations with low confidence ratings were ignored. As anticipated, the algorithms minimized false positives at the expense of false negatives, resulting in relatively high values for precision and relatively low values for recall (Table 2). Accuracy ratings are included but because there was a significant class imbalance for each of the behaviors it is potentially misleading as a performance measure. (Of 56 participants, manual coding found 48 *tryers*, 8 *keyworders*, and 21 *morers*.) F-scores, the harmonic mean of precision and recall, provide an indication of whether the balance of the two is reasonable. The $F_{0.5}$ score is considered more appropriate because it weights recall lower than precision (by attenuating the influence of false negatives) which is consistent with the design objective of avoiding false positives ahead of reducing false negatives.

Table 2. Algorithm performance

	Accuracy	Precision	Recall	F_1	$F_{0.5}$
Tryer	0.702	0.919	0.708	0.800	0.867
Keyworder	0.912	0.714	0.625	0.667	0.694
Morer	0.807	1.000	0.421	0.593	0.784

2.3 Application

The algorithms were implemented as software agents which parse and analyze the conversation in real-time. Another agent used this information to direct the CA to inject interventions into conversation and/or modify the CA's responses. The multiple agent framework [10] deployed the following interventions to improve engagement:

***Tryer* Intervention:** The expected outcome is an improvement in the pedagogical utility of the experience, i.e., the delivery of more educational (Freud related) content. The user is conversationally engaged, but the CA is unable to deliver useful information, due to typos, grammar, or poor CA performance. The strategy is to take some control of the conversation and provide content, using a conversational approach. For example, narrowing down the area of interest, and then suggesting a topic.

***Keyworder* Intervention:** The desired outcome is to boost conversational engagement by encouraging a more conversational approach. Students who exhibit *keywording* behavior may receive the domain content, but are not taking full advantage of the capabilities of the interface, including the option to delve deeper into topics, change topics, or ask analytical questions. If *keyworder* behavior is detected, the CA reminds students they can explore a topic using phrases such as "Tell me more about…".

***Morer* Intervention:** The goal is for students to use conversational acts to drive how content is delivered rather than relying on the systematic, ordered output of the narratives of each topic. This serves to involve higher cognitive processes that consider different branches in the structure of the topics. *Morer* behavior is similar to pressing a "next" button, so there is little incentive for the student to use this tool over reading a text book. The CA reminds students they can branch to other topics ("Tell me about") or return to a topic ("Tell me more about…").

3 User Testing

A pilot study was conducted to verify the effectiveness of introducing interventions triggered by real time identification of user interaction behaviors. 56 volunteer student participants (13 men, 43 women, aged 18 to 63 with 63% under the age of 32) chatted with the agent-based Freudbot CA described in Sect. 2 and [10]. Interactions were carried out remotely by web interface. Participants were required to chat with Freudbot for at least 10 min. An online survey collected feedback on the experience.

3.1 Results

Student responses were compared to confirm the value of using student conversational behaviors to initiate appropriate interventions to improve interaction. Three survey questions related to the users' perceptions of CA performance along three conversational control dimensions: 1. *How appropriate or useful were the suggestions regarding conversing with Freudbot?* (Intervention Rating); 2. *Overall, how would you rate Freudbot's response when he did not appear to understand?* (No-match Response); 3. *How would you rate Freudbot's choice of topics?* (Topic Suggest Rating).

Topic suggestions, and responses when user input is not understood, are adjusted when one of the behaviors is detected, along with conversational suggestions. We compared ratings for each conversational control mechanism against the participants' declaration of whether they would choose to talk to Freudbot again. A Mann-Whitney U test indicated that participants who would chat again rated conversational control measures higher than those who would not (Table 3). This suggests these measures, and therefore the interventions that affect them, are important to the user experience.

Table 3. Conversational control measures vs Chat Again - Mann Whitney U test

	Chat Again	N	Mean rank	U	p
Intervention rating	No	15	10.43	36.50	.000
	Yes	17	21.85		
	Total	32			
No-match response	No	27	21.67	207.00	.001
	Yes	29	34.86		
	Total	56			
Topic suggest rating	No	23	15.67	84.50	.000
	Yes	25	32.62		
	Total	48			

We looked next at the relationship between the interventions and perception of learning utility. Two questions related to the participants' perception: 1. *How useful is this activity for learning information about Sigmund Freud?*; 2. *How useful is this activity for remembering information about Sigmund Freud?*

These responses were compared to participants' rating of the interventions. A Spearman's correlation test showed a significant correlation between those who rated the intervention as good/appropriate and those who rated the value of the CA high for learning (r = .38, p < .05) and remembering (r = .52, p < .005).

4 Discussion and Conclusion

The performance of the algorithms in recognizing three conversational behaviors related to student interaction with a CA is comparable to that of human judgement. We demonstrated that these algorithms can advise a CA, using an agent-based framework, to initiate interventions.

Early results indicate the quality of these interventions influences whether students find the CA to be useful for learning and remembering, and affect their overall rating of the experience. A correlation was found between participants who rated interventions as useful and/or appropriate and those who stated they would be willing to chat with the CA again. Similar correlations to chat again were found for ratings of how the CA responded to no-match cases, and the selection of topics suggested by the CA to keep the conversation going.

We relied on participants' declaration of whether they would choose to chat with this CA again as a measure of overall satisfaction with the experience. This was justified by a high degree of correlation between this measure and all other questions related to overall rating of the CA. However, some participants who were familiar with chat bots found Freudbot to be lacking by comparison, which affected satisfaction ratings. Other comments indicated they saw no potential in the concept in general.

The pilot study provided data on how students interact with a CA and how it is perceived. The next step is to design of a controlled experiment to compare learning outcome measures for the agent-based system against one with no interventions.

References

1. Kerry, A., Ellis, R., Bull, S.: Conversational agents in E-Learning. In: Allen, T., Ellis, R., Petridis, M. (eds.) Applications and Innovations in Intelligent Systems, vol. XVI, pp. 169–182. Springer London (2009). https://doi.org/10.1007/978-1-84882-215-3_13
2. Szafir, D., Mutlu, B.: Pay attention!: designing adaptive agents that monitor and improve user engagement. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 11–20. ACM, New York (2012)
3. Nakano, Y.I., Ishii, R.: Estimating user's engagement from eye-gaze behaviors in human-agent conversations. In: Proceedings of the 15th International Conference on Intelligent User Interfaces, pp. 139–148. ACM, New York (2010)
4. Xu, Q., Li, L., Wang, G.: Designing engagement-aware agents for multiparty conversations. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 2233–2242. ACM, New York (2013)
5. Asteriadis, S., Karpouzis, K., Kollias, S.: Feature extraction and selection for inferring user engagement in an HCI environment. In: Jacko, J.A. (ed.) HCI 2009. LNCS, vol. 5610, pp. 22–29. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-02574-7_3
6. Paquette, L., Baker, R.S.J.D., Sao Pedro, M.A., Gobert, J.D., Rossi, L., Nakama, A., Kauffman-Rogoff, Z.: Sensor-free affect detection for a simulation-based science inquiry learning environment. In: Trausan-Matu, S., Boyer, K.E., Crosby, M., Panourgia, K. (eds.) ITS 2014. LNCS, vol. 8474, pp. 1–10. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-07221-0_1
7. Wen, M., Yang, D., Rose, C.P.: Linguistic reflections of student engagement in massive open online courses. In: Eighth International AAAI Conference on Weblogs and Social Media (2014)
8. Turney, P.D., Neuman, Y., Assaf, D., Cohen, Y.: Literal and metaphorical sense identification through concrete and abstract context. In: Proceedings of the 2011 Conference on the Empirical Methods in Natural Language Processing, pp. 680–690 (2011)

9. Heller, R., Procter, M.: Animated pedagogical agents: the effect of visual information on a historical figure application. Int. J. Web-Based Learn. Teach. Technol. **4**, 54–65 (2009)
10. Procter, M., Lin, F., Heller, R.: Improving conversation engagement through data-driven agent behavior modification. In: Khoury, R., Drummond, C. (eds.) AI 2016. LNCS (LNAI), vol. 9673, pp. 270–275. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-34111-8_33

Extraction of Relevant Resources and Questions from DBpedia to Automatically Generate Quizzes on Specific Domains

Oscar Rodríguez Rocha(✉), Catherine Faron Zucker, and Alain Giboin

University Côte d'Azur, CNRS, Inria, I3S, Nice, France
{oscar.rodriguez-rocha,alain.giboin}@inria.fr, faron@unice.fr

Abstract. Educational quizzes are useful not only to evaluate or test the knowledge acquired by a learner, but also to help her/him to deepen knowledge about a specific domain or topic in an informal and entertaining way. The production of quizzes is a time-consuming task that can be automated by taking advantage of existing knowledge bases available on the Web of Linked Open Data (LOD). However, automatically extracting from the LOD a knowledge graph composed by the information of a set of resources which are relevant to a given specific domain or topic, is a crucial phase for the automatic generation of quizzes.

To address this issue, we propose a heuristic that extracts from DBpedia a set of resources related to a given specific domain. Such heuristic has been implemented and used for the automatic generation of quizzes in the geography and privacy domains. We report a comparative user evaluation of it.

1 Introduction

Educational quizzes are useful not only to evaluate or test the knowledge acquired by a learner, but also to help her/him to deepen knowledge about a specific domain or topic in an informal and entertaining way. The so-called Semantic Web introduces semantics into the Web to extend its capabilities. It relies on the publication of structured data which can be viewed as a global giant knowledge graph and on ontologies which capture the relations among concepts [1] and provide the semantics that machines can understand and process. The enormous and continuous growth of this global knowledge graph makes it a rich source of structured data. As discussed in [2], the generation of quizzes *is a time-consuming task that can be automated by taking advantage of existing knowledge bases available on the Semantic Web*, for this, authors proposed an approach that relies on the work of [3] on the generation of multiple choice questions from domain ontologies through queries. However, some of those available knowledge bases (like DBpedia) may contain resources from different domains or topics, thus the automatic extraction of a knowledge graph which contains resources relevant to a specific domain or topic is a crucial phase for the automatic generation of quizzes. By *relevant resource*, we mean a resource whose

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 380–385, 2018.
https://doi.org/10.1007/978-3-319-91464-0_43

information or content is related to a specific domain and therefore it is likely to be used for the generation of questions of such domain.

The research work presented in this paper addresses the research question: *How to select a set of resources relevant to a topic or domain from a knowledge base, and extract a knowledge graph in order to be able to automatically generate quizzes from it?*

For this, we have focused our study on the DBpedia knowledge base and we have considered the definition of a topic or a domain as an input set of keywords in natural language. As a result, we propose a heuristic that selects a set of DBpedia resources that are relevant to the specified domain to generate a specific knowledge graph with their structured information, from which the questions of a quiz are generated. This heuristic finds an initial set of relevant resources through a process of entity linking and then enriches them with additional resources that are obtained through a filtering process applied to their *wikilinks*. We have carried out a comparative evaluation of this heuristic against the baseline (a heuristic that applies an entity linking process to the keywords that define a domain) in terms of relevance. In addition, we have evaluated the relevance of the questions generated from the knowledge graphs extracted by both heuristics. By *relevant question*, we mean a question that requests information about a specific domain or allows to verify knowledge about it.

The remainder of this paper is structured as follows: In Sect. 2, we present and detail our proposed heuristic. In Sect. 3, we describe the implementation and evaluation of the proposed heuristic. In Sect. 4, we present the related works. Finally, conclusions and future work are presented in Sect. 5.

2 Proposed Approach

The selection of resources that are relevant to a specific domain or topic from a dataset is the basis for automatically generating useful quizzes for the learners. Since the knowledge bases on the Semantic Web use different ontologies and ways of structuring their data, we decided to focus our study on DBpedia since it is widely used and provides a large amount of resources from different domains.

We have considered that the simplest way to specify the domain or topic for which we want to generate quizzes, is to start with a set of keywords. From this set of keywords, in order to extract a subgraph from DBpedia with resources relevant to the described domain, we have designed our proposed heuristic, whose objective is (1), to be able to discover more relevant resources from DBpedia than with the baseline, and (2) to limit the number of non-relevant resources that may be discovered. This heuristic is inspired by the work described in [4]. We consider as a baseline, an entity linking process applied to the domain-specific keywords, such as applying DBpedia Spotlight[1] to extract a set of resources and their RDF triples to create a knowledge graph. Theoretically, it selects resources with a greater precision and lesser recall.

[1] http://www.dbpedia-spotlight.org.

We will report on two different experiences: In the first one, the domain is initially specified through a set of 107 keywords resulting from a process of manual annotation of a representative set of 126 questions, extracted from the famous French game *"Les Incollables"*[2], about geography. In the second experience, the domain is initially specified through a set of 240 keywords extracted manually by an educational engineer, from resources of a MOOC about privacy[3].

2.1 Named Entity and Filtered Category Extraction

Our proposed heuristic extracts a set of DBpedia named entities R from the set of keywords that describe the targeted domain and enrich them with the *wikilinks* of the those extracted resources having relevant categories. A set of DBpedia categories C, is built with the result of a dedicated SPARQL query on DBpedia searching for the value of the *dcterms:subject* property or *dcterms:subject/skos:broader* property path of each resource in R. It considers the most relevant categories of the domain C_{topK} to filter the *wikilinks*.

We define C_{topK} as a subset of C that has k elements, which are the most relevant categories having the more related resources. The value of the number of categories k is determined by a manual analysis of the relevance of the categories with respect to the domain, allowing to discard some categories that may not be relevant to the targeted domain. For the domains of geography and privacy, the value of the threshold k was empirically fixed to 11 and 10 respectively.

As for the above-described heuristics, the set of triples describing the resources in R are stored in a named graph NG. Theoretically, this heuristic selects resources with a greater recall and lesser precision.

3 Empirical Validation of the Proposed Approach

We have applied the above described heuristics to the sets of keywords that define the two domains considered: geography and privacy.

To validate our proposed approach to extract a knowledge graph from DBpedia relevant to a given a specific domain to generate quizzes, we first evaluated the relevance of the resources selected by the baseline and our heuristic, then we measured the relevance of the questions generated from the different generated knowledge graphs.

3.1 Evaluation of the Relevance of the Selected Resources

For the domain of *Privacy*, we have asked three students registered to the MOOC about privacy (who were also familiar with DBpedia), to evaluate the relevance of the resources selected. The list of all the resources selected by the baseline and our heuristic (and merged to avoid duplicates) was presented to them into

[2] http://www.lesincollables.com.

[3] https://www.fun-mooc.fr/courses/course-v1:inria+41015+session01/about.

a spreadsheet. Each resource was evaluated by the students on a scale of 1 (not at all relevant) to 5 (very relevant). Again, considering that a relevant resource is a resource related to the specific domain and therefore likely to be used in the generation of a question.

Once the relevance of the resources was evaluated, we calculated the precision and the recall of the baseline and our heuristic per user. We defined them as the proportion of relevant resources among all the resources generated by a given heuristic and the proportion of relevant resources generated by a given heuristic among all the relevant resources generated by any of the two heuristics, respectively.

According to the previous defined scale of relevancy, we considered that a resource is sufficiently relevant if its score is greater than or equal to 3.

Finally the average precision and recall (considering the three evaluators) are reported in Table 1. For the domain of *Geography*, we have asked three school teachers to evaluate the relevance of the resources obtained. Similarly to the previous experimentation, a list of resources was presented to the evaluators in a spreadsheet, to be evaluated on a scale of 1 to 5.

The average precision and recall (considering also the three evaluators) are also reported in Table 1.

Table 1. Precision and recall of each heuristics by domain

	Privacy		Geography	
	BL	H	BL	H
Precision	0,567668401	0,572322652	0,957446809	0,906040268
Recall	0,34855581	0,712087542	0,25862069	0,775862069

3.2 Evaluation of the Relevance of the Generated Questions

After having conducted the evaluation of relevance of the resource selected by each proposed heuristic, we have applied the quiz generation techniques proposed in [2], to the DBpedia subgraphs generated by the baseline and our proposed heuristic.

Finally, we have asked the evaluators to evaluate the relevance of the questions according to their corresponding domain, generated from each heuristic (on a scale of one to 5, where 5 is the most relevant). For this, they have been provided with a list of 100 questions extracted randomly from the subgraph created for each ontology and each domain.

The results of this evaluation are shown below in Table 2.

3.3 Discussion of the Results

The results of the evaluation of the relevance of resources for the *Geography* domain are similar to those of the *Privacy* domain: our proposed heuristic

Table 2. Average relevance of the questions per domain and subgraph

	SubGraph BL	SubGraph H
Geography	3.59	3.5
Privacy	4.2	3.62

obtains the highest recall while keeping an accuracy not so inferior with respect to that of the baseline. Thus we can expect that our proposed heuristic is more adequate than the baseline to generate the knowledge graph from which to generate questions.

This was confirmed by analyzing the relevance of the questions generated from the two knowledge graph: the questions generated from the knowledge graph generated by the baseline were those with the highest relevance. Nevertheless, the difference in relevance with respect to the questions generated from the knowledge graph generated by our heuristic is not so great, thus our heuristic can be considered as an excellent option since it is able to discover a larger amount of related resources and therefore of questions with greater novelty.

4 Related Work

In [5], the authors propose *an approach to identify a minimal domain-specific subgraph by utilizing statistic and semantic-based metrics*. This approach targets DBpedia as a knowledge base and focuses on identifying entities and relationships strongly associated with the domain of interest. They describe the domain of interest through a main entity that represents it. In contrast, we present an approach to describe a domain in a more complete and specific way.

In [4,6,7], the authors describe an approach to exploit semantic relations stored in the DBpedia dataset to extract and rank resources related to the user context given by the keywords she enters in a search engine to formulate her query. Compared to this related work, our approach seeks to rely only on DBpedia and does not consider external sources of unstructured knowledge. Additionally, it does not currently consider estimating the *strength of the connection* between two resources linked through a *wikilink* property.

Authors in [8], present an approach to generate questions from the LOD for the History domain. In contrast to this work, our approach allows to create knowledge graphs about domains that can be defined in a more granular and specific way through keywords, and we do not rely on the use of DBpedia classes, but on entity linking combined with a process to discover related resources.

In [9], the authors present a list of state of the art works about resource discovery and graph exploration.

5 Conclusions and Future Work

In this paper, we have presented an approach for the extraction of relevant knowledge graphs from DBpedia in order to automatically generate quizzes on specific domains. For this we have proposed and detailed a heuristic that we have further evaluated with two real life domains and educational contexts: quizzes to enrich a MOOC on privacy and quizzes to populate a serious game on geography. We have applied techniques of automatic generation of quizzes from the resulting knowledge graphs to understand the impact of the heuristics chosen to generate the input knowledge graph on the generated quizzes. The baseline heuristic consists in considering the graph of the descriptions of the named entities extracted in DBpedia from a set of keywords can be refined. The experiments showed that (1) Enriching the knowledge graph with semantically related DBpedia resources enables to increase the number of generated questions, and (2) Ranking the candidate related resources by their degree of relevancy to the domain enables to maintain the precision of the generation of questions.

As future work we will seek to improve our heuristic, by considering the participation of an expert user to the process of validating the automatically extracted categories and eventually the ranking of resources. This should improve the relevance of the generated quizzes to the domain or topic considered. Finally, we plan to evaluate the heuristic with other domains or topics.

References

1. Berners-Lee, T., Hendler, J., Lassila, O., et al.: The semantic web. Sci. Am. **284**(5), 28–37 (2001)
2. Rodriguez Rocha, O., Faron Zucker, C.: Automatic generation of educational quizzes from domain ontologies. In: EDULEARN17 Proceedings. 9th International Conference on Education and New Learning Technologies, IATED, pp. 4024–4030 (2017)
3. Papasalouros, A., Kanaris, K., Kotis, K.: Automatic generation of multiple choice questions from domain ontologies. In: Nunes, M.B., McPherson, M., (eds.) e-Learning, IADIS, pp. 427–434 (2008)
4. Mirizzi, R., Ragone, A., Di Noia, T., Di Sciascio, E.: Semantic tags generation and retrieval for online advertising. In: Proceedings of the 19th ACM International Conference on Information and Knowledge Management. CIKM 2010, New York, NY, USA, pp. 1089–1098. ACM (2010)
5. Lalithsena, S., Kapanipathi, P., Sheth, A.: Harnessing relationships for domain-specific subgraph extraction: A recommendation use case. In: 2016 IEEE International Conference on Big Data (Big Data), pp. 706–715 (2016)
6. Mirizzi, R., Ragone, A., Di Noia, T., Di Sciascio, E.: Semantic tag cloud generation via DBpedia. In: Buccafurri, F., Semeraro, G. (eds.) EC-Web 2010. LNBIP, vol. 61, pp. 36–48. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15208-5_4
7. Mirizzi, R., Ragone, A., Di Noia, T., Di Sciascio, E.: Semantic wonder cloud: exploratory search in DBpedia. In: Daniel, F., Facca, F.M. (eds.) ICWE 2010. LNCS, vol. 6385, pp. 138–149. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-16985-4_13
8. Jouault, C., Seta, K., Hayashi, Y.: Content-dependent question generation using lod for history learning in open learning space. New Gener. Comput. **34**(4), 367–394 (2016)
9. Figueroa, C., Vagliano, I., Rodríguez Rocha, O., Morisio, M.: A systematic literature review of linked data-based recommender systems. Concurr. Comput. Pract. Exp. **27**, 4659–4684 (2015)

A Planning-Based Approach to Generating Tutorial Dialog for Teaching Surgical Decision Making

Narumol Vannaprathip[1](✉), Peter Haddawy[1], Holger Schultheis[2], Siriwan Suebnukarn[3], Parichat Limsuvan[3], Atirach Intaraudom[1], Nattapon Aiemlaor[1], and Chontee Teemuenvai[1]

[1] Faculty of ICT, Mahidol University, Nakhon Pathom, Thailand
{narumol.van,atirach.int,nattapon.aie, chontee.tee}@student.mahidol.ac.th, peter.had@mahidol.ac.th

[2] Bremen Spatial Cognition Center, University of Bremen, Bremen, Germany
schulth@informatik.uni-bremen.de

[3] Faculty of Dentistry, Thammasat University, Pathum Thani, Thailand
{ssiriwan,lparicha}@tu.ac.th

Abstract. Teaching surgical decision making aims at enabling students to choose the most appropriate action relative to the patient's situation and surgical objectives. This requires a deep understanding of causes and effects related to the surgical domain as well as being aware of key properties of the current situation. To develop an intelligent tutoring system (ITS) for teaching situated decision making in the domain of dental surgery, in this paper, we present a planning-based representation framework. This framework is capable of representing surgical procedural knowledge with respect to situation awareness and algorithms that utilize the representation to generate rich tutorial dialog. The design of the tutorial dialogs is based on an observational study of surgeons teaching in the operating room. An initial evaluation shows that generated interventions are as good as and sometimes better than those of experienced human instructors.

Keywords: Surgical decision making · Situation awareness · Planning
Pedagogical intervention · Knowledge representation

1 Introduction

Teaching surgical decision making aims at providing students with the knowledge and skills to choose the most appropriate actions relative to the surgical objectives and patient state in routine and non-routine situations. This requires giving students a deep understanding of causal relations in the surgical domain as well as the ability to monitor important cues and interpret their meaning. This combination of skills is termed situation awareness (SA), which is "the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future" [1]. While a few simulators exist for teaching surgical decision making, none focuses on teaching SA skills. In previous work, we showed how

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 386–391, 2018.
https://doi.org/10.1007/978-3-319-91464-0_44

the Planning Domain Definition Language (PDDL) may be used to represent actions in surgical procedures and elements of situation awareness [2]. PDDL can be used to represent important aspects of surgical actions including action parameters and conditional effects. Its domain axioms can be used to model the process of comprehension of perceived facts. The elements of the SA framework are clearly distinguished, enabling reasoning at a level needed for generating causal explanations [3]. In this paper we extend the previous work by adding a representation of how actions are structured to form a surgical plan, including its hierarchical structure. We extend the previous plan projection engine to generate plan traces and show how this enables additional important teaching interventions identified in an observational study, in particular questions and explanations concerning causal relations that span more than one action. Initial evaluation shows generated interventions are as good as and sometimes better than those of experienced human instructors.

2 Surgical Procedure and Surgical Training

A surgical procedure is a hierarchical plan that consists of a sequence of tasks and subtasks, and on the lowest level consists of a sequence of individual actions. Each task/action is dependent on the current status of the patient and the effects of the previously executed task/action, respectively. A surgical procedure requires a student to carry out surgical tasks and during training a surgical expert may intervene in different ways such as providing feedback and asking questions. These interventions are often related to cause/effect relations. To understand the exact nature of and reasons behind expert interventions, an observational study of nine teaching sessions of endodontic treatment and interviews with dental instructors was conducted. Sixteen teaching strategies were identified. Implementing these strategies requires a knowledge representation formalism that (a) represents key elements in the surgical procedure: actions with parameters and conditional effects, (b) represents the components in the SA framework including perception, comprehension, and projection, (c) represents information relevant to the procedure e.g., patient state information, (d) allows integrating available information to derive new information (e.g., integrating perceptual facts with the user's action to derive action effects), and (e) supports generating tutorial dialog.

3 Related Work

A few ITSs use representations based on AI planning languages. Annie [4] uses a STRIPS-like language to compute a directed graph that represents the space of all possible plans from a given state in its game world. Annie uses this to prioritize and sequence its strategies for guiding students to obtain the requisite knowledge. Roman Tutor [5] is an intelligent simulator aimed to train an astronaut to operate an articulated robot arm mounted on the international space station. PDDL is applied to represent continuous shots in the camera planner for generating demonstrations. Students can ask what-if and why-not questions and the tutor can provide illustrations and explanations if the actions are not appropriate.

Critiquing is one important type of tutoring feedback which has been extensively studied in the area of plan critiquing. ATTENDING [6] critiques preoperative anesthetic management plans by concatenating strings from an Augmented Transition Network. Trauma-TIQ [7] applies a pure rule-based approach to critique a physician's plan. There is no explicit representation of a surgical procedure nor distinguished types of plan components.

Computer Interpretable Guidelines (CIG) represent medical knowledge to share across medical institutions for decision support [8]. Of the numerous CIGs available, Asbru [9] is the closest to our work. It represents medical knowledge as a skeletal plan with action effects but not conditional effects. This is due to the fact that Asbru, like all CIGs, is designed for specification of prescriptive guidelines and thus just specifies the conditions under which an action achieves its desired effect.

While a variety of representations have been used for ITSs, none of them fulfills the requirements arising from the properties of surgical procedures and training (see above). In rule-based models and model tracing [10], components of the surgical procedure and SA framework can be represented via a set of facts, and their relations to the rules. These facts do not explicitly represent components of surgical procedure and SA framework. The constraint-based models [10] focus on the correctness of the solution represented in a form of constraints rather than the sequence of performed actions; consequently, action representation that is important for the surgical procedure is not explicitly found nor conditional action effects. Even though the constraint-based model was demonstrated to represent the operator rules in model tracing via constraints [11], in the aspect of surgical procedure, their notions are not distinguished to different components in the surgical procedure, in particular an action representation, and SA framework. The qualitative models [12] work well for describing indirect effects – changes of objects after an action is executed. Work on qualitative reasoning about actions has integrated TPLAN and STRIPS action representations with the qualitative models [13, 14].

4 Automated Tutor/System Implementation

4.1 Knowledge Representation

The PDDL representation from our previous work [2] of domain rules, action description with conditional effects, and elements of SA framework does not provide a plan that formulates of the structure of actions. We utilize the NPDDL plan representation [15] to represent a sequence of tasks/actions in a surgical plan. This is a control structure for the automated tutor to validate student's actions with respect to given steps in a procedure. A procedure or a plan is divided into a sequence of sub-plans. Figure 1 illustrates the structure of partial root canal treatment plan and a portion of its sub-plan - the rubber dam application. A sub-plan can be optional for a particular condition (see: `optional` at the `local_anes_plan`). The sub-plan is composed of actions arranged into a sequence. Each action is described with a set of desired outcomes and one of them refers to the main objective of the action.

```
(define(plan root_canal_treatment_plan)
(:domain root_canal_treatment)
(:body (sequence
  (subplan(local_anes_plan)
      :optional(diagnosis pulp_necrosis))
  (subplan(rubber_dam_plan)))))
```

```
(define (subplan rubber_dam_plan )
(:plan root_canal_treatment_plan)
(:domain root_canal_treatment) (:body (sequence
(action (insert_rubber_dam):desired_outcome
        (AND WORKING_TOOTH_SEPARATED) :main
            (NOT(PATIENT_RISK_TO_FAINT)))))
```

Fig. 1. The partial root canal treatment plan and its sub-plan – rubber dam application

4.2 Plan Projection Engine and Graph Representations

The patient's initial conditions and their comprehension facts derived from domain rules are formulated as an initial state. When the action is performed, the plan projection engine uses forward chaining to project effects from the action conditions against the current state and formulates a new state to evaluate if the executed action satisfy the desired outcomes. A surgical procedure graph is created to represent the surgical procedural domain knowledge illustrating the SA elements and the plan components. The student-executed action is represented as a subset of the surgical procedure graph (see checked nodes). The projections of one executed action become part of the perception of the next step. Figure 2 illustrates how an effect of an action "`Insert rubber dam`" is related to the next action step "`Drill to pulp chamber`".

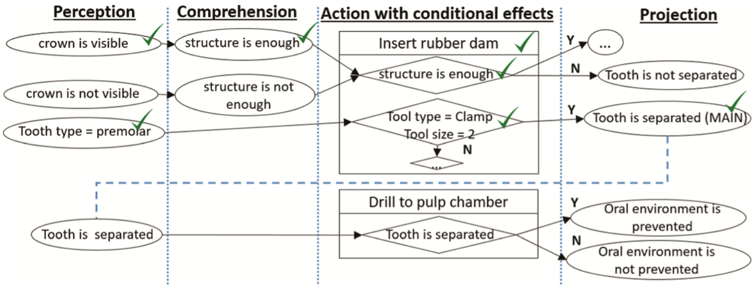
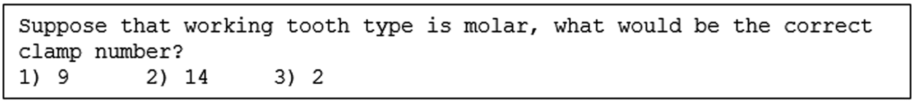


Fig. 2. Partial surgical procedure graph of two adjacent steps with a student solution

4.3 Tutorial Intervention Generation

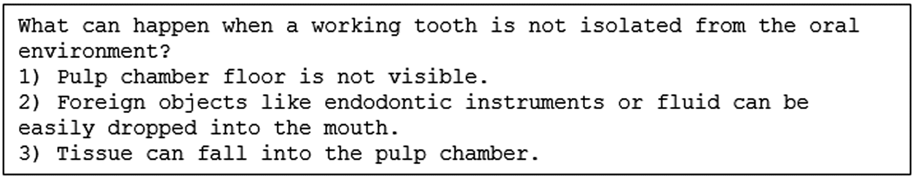
The surgical procedure graph and its subset the student performance graph were utilized to generate tutorial interventions. We present two dialogues: First, a question about a hypothetical situation is generated to provide a broader understanding of the working procedure. For example, when the student successfully inserted the rubber dam using clamp number 2 for a premolar tooth, the automated tutor looks for another similar action condition node i.e., "`tooth type = molar, tool type = Clamp, and tool size = 14`". Among these retrieved action conditions, the duplicated conditions such as "`tool type`" are cleaned out. The perceptual information "`tooth type`" becomes a part of the question. The student action detail "`tool size`" becomes the answer. The set of possible answers is determined by other available values of tool size in the domain. Figure 3 shows a sample of a hypothetical question of applying a rubber dam.



```
Suppose that working tooth type is molar, what would be the correct
clamp number?
1) 9      2) 14     3) 2
```

Fig. 3. A question about a hypothetical situation

A student may correctly execute a task despite not fully understanding the rationale of the task. A question about future consequence(s) is generated to raise an awareness. For example, when the student failed to answer the question about the objective of the task "Why did you insert the rubber dam?" the automated tutor negates the current projected desired outcome "`tooth is separated`" as a key to search for action condition nodes in the later steps to generate this question. The projection node of the selected action condition "`oral environment is not isolated`" is the answer, alternative choices are projection nodes randomly selected from the same step. Figure 4 shows a question for raising awareness about future consequences.



```
What can happen when a working tooth is not isolated from the oral
environment?
1) Pulp chamber floor is not visible.
2) Foreign objects like endodontic instruments or fluid can be
easily dropped into the mouth.
3) Tissue can fall into the pulp chamber.
```

Fig. 4. A question about future consequence(s)

5 Evaluation

We asked four endodontic clinical instructors, who had at least five years of clinical experience to provide tutoring feedback for five different situations related to the stages of selecting the local anesthesia solution and providing the rubber dam application. These two steps are selected because the rubber dam application is mandatory while selecting the local anesthesia can be optional. Five different situations include: (a) the student selects the local anesthesia solution for a patient with pulpitis; (b) the student prepares access without inserting the required rubber dam; (c) student correctly inserts the rubber dam but fails to answer a question about the task objective "Why do you insert the rubber dam?"; (d) the student correctly inserts the rubber dam and successfully answers the same question from the situation (c); and (e) the student inserts the rubber dam using the incorrect clamp number. The tutorial feedback from these four instructors and the generated feedback from the automated tutor were blindly rated for appropriateness of interventions using a 5-point Likert scale and commented by two endodontic experts, who had more than 10 years of clinical experience and are responsible for evaluating the teaching performance of human tutors. A mean score for each instructor over these five situations was computed. The mean scores of the human instructors are 4.8, 4.4, 4, and 3.6. The mean score of the automated tutor is 4.4, which is larger than the average human score of 4.24 and close to the best human instructor.

6 Conclusion and Future Work

We have provided a representation of surgical plans using PDDL and NPPDL. We use this to generate plan traces that are used by a pedagogical module to generate a wide variety of teaching dialog observed in teaching sessions in the operating room. Initial evaluation indicates that the generated teaching interventions are as good as those of human tutors. The evaluation also identified some dialog that can be improved such as generation of good multiple choice questions. Future work will focus on improving the generation of multiple choice questions, interfacing the pedagogical engine with the simulator, and conducting a larger scale and more comprehensive evaluation.

Acknowledgments. This research project was partially supported by the Faculty of Information and Communication Technology, Mahidol University.

References

1. Endsley, M.R.: Toward a theory of situation awareness in dynamic systems. Hum. Factors J. Hum. Factors Ergon. Soc. **37**, 32–64 (1995)
2. Vannaprathip, N., Haddawy, P., Schultheis, H., Suebnukarn, S.: Generating tutorial interventions for teaching situation awareness in dental surgery – preliminary report. In: Phon-Amnuaisuk, S., Ang, S.-P., Lee, S.-Y. (eds.) MIWAI 2017. LNCS (LNAI), vol. 10607, pp. 69–74. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-69456-6_6
3. Mccluskey, T.L.: PDDL: a language with a purpose? In: 13th International Conference on Automated Planning & Scheduling, ICAPS 2003. AAAI Press, Trento (2003)
4. Thomas, J.M., Young, R.M.: Annie: automated generation of adaptive learner guidance for fun serious games. IEEE Trans. Learn. Technol. **3**, 329–343 (2010)
5. Beghith, K., Kabanza, F., Nkambou, R., Khan, M., Hartman, L.: Roman tutor: a robot manipulation tutoring simulator. DEM **20** (2005)
6. Miller, P.L.: Critiquing anesthetic management: the “ATTENDING” computer system. Anesthesiology **58**, 362–369 (1983)
7. Gertner, A.S.: Plan recognition and evaluation for on-line critiquing. User Model. User-Adapt. Interact. **7**, 107–140 (1997)
8. De Clercq, P.A., et al.: Approaches for creating computer-interpretable guidelines that facilitate decision support. Artif. Intell. Med. **31**, 1–27 (2004)
9. Miksch, S., et al.: Asbru: a task-specific, intention-based, and time-oriented language for representing skeletal plans. In: 7th Workshop on Knowledge Engineering Methods and Languages, pp. 1–25 (1997)
10. Nkambou, R., Bourdeau, J.: Advances in Intelligent Tutoring Systems. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-14363-2
11. Kodaganallur, V., et al.: A comparison of model-tracing and constraint-based intelligent tutoring paradigms. Int. J. Artif. Intell. Educ. **15**, 117–144 (2005)
12. Forbus, K.D.: Qualitative process theory. Artif. Intell. **24**, 85–168 (1984)
13. Hogge, J.C.: Compiling expressed plan operators from domains recess theory in qualitative to TPL. In: AAAI, pp. 229–233 (1987)
14. Forbus, K.D.: Introducing actions into qualitative simulation (1988)
15. Bertoli, P., et al.: Extending PDDL to nondeterminism, limited sensing and iterative conditional plans. In: ICAPS 2003 Workshop on PDDL (2003)

Supporting Multiple Learning Experiences on a Childhood Vocabulary Tutoring Platform

Aditya Vempaty(✉), Tamer Abuelsaad, Allison Allain, and Ravi Kokku

IBM Watson Education and Research, Yorktown Heights, USA
{avempat,tamera,acallain,rkokku}@us.ibm.com

Abstract. We present a unified learner modeling approach in a childhood vocabulary tutoring platform that enables learning continuum across multiple learning experiences. By decoupling experiences from learner modeling, multiple learning experiences can be developed independently making learning approaches scalable in an inherently diverse setting: each child, his/her family, environment, geography, cultural differences can all make each learner really unique. By understanding the information-theoretic equivalence of different assessment types, and mapping different play and learning activities to one or more of these types, we can enable a rapid convergence of the learner model to better represent a young learner's knowledge. More interestingly, when normalized, different assessment types converge at different levels of the normalized score.

1 Introduction

Childhood vocabulary learning happens organically across many experiences of a young learner. As formal schooling begins, vocabulary acquisition continues implicitly while listening and reading quality texts, and explicitly through word instruction. Research suggests a correlation between early vocabulary acquisition and ongoing academic success [1–3,8], and that the highest rate of vocabulary development occurs during early childhood years [6]. Unfortunately, some children arrive to school with less vocabulary knowledge, which could hinder their ongoing academic achievement [7]. Despite this importance, researchers found very few effective programs dedicated to early vocabulary instruction [9].

To this end, we enable an intelligent vocabulary tutoring platform, over which one can build (i) multiple mobile applications, toys, and games for explicit vocabulary instruction, (ii) automated assessments to continuously gauge a child's learning (while being organically or explicitly exposed to a variety of learning experiences), and (iii) learner modeling that closely represents a child's understanding of different word concepts, and updates the models based on information from all learner-platform interactions. While designing the platform, assessments are classified into different types for understanding a learner's knowledge. We identify that the learner knowledge information *gained* by a system from

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 392–398, 2018.
https://doi.org/10.1007/978-3-319-91464-0_45

a child's assessment response depends on the assessment type. We develop an entropy-normalization method to update a unified model of a child's understanding of a word, and enable assessments of multiple types.

Prentzas surveys popular educational technology tools for early childhood learning [10]. Accomplishing individualization has been a key enthusiasm among engineers of interactive learning environments and Intelligent Tutoring Systems (ITSs). To accomplish one-on-one individualized instruction, fine-grained adaptation is required, which requires precise modeling of learners' knowledge. Desmarais and Baker [5] identify that the isolated nature of learner models is a limitation in many widely used learning environments. For instance, rich representations of learners' knowledge are created and refined by teachers but discarded at the end of the school year. A continuum of these learner models could be extremely useful in future learning and in co-occurring classes, while also enabling measurement and representation of implicit knowledge acquisition in organic settings. Further, with increasing online education, a learner could encounter content about a particular concept multiple times. Unified and shared learner models would help prevent repetition, and only target areas of weaknesses.

2 Vocabulary Tutoring Platform Architecture

Figure 1 gives a high level overview of the architecture. At the base of the tutor platform are knowledge maps that are words and their relationships, attributes such as parts of speech, definitions and usage of the words, and links to learning and assessment content for each word. At the core of the platform is the learner model that represents the system's confidence of a learner's understanding of a word; the confidence is derived out of the learner's assessment responses over time. The learning content is personalized based on the learner model to focus remediations around only concepts that are weaknesses.

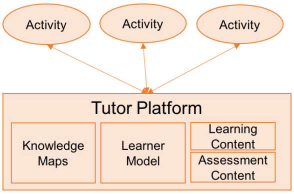


Fig. 1. Tutor platform with multiple activities

Young learners learn in a variety of settings and with a number of mobile applications, toys, and games. Hence, a tutoring platform enabling continuous learning via these activities, while maintaining a consistent view of the child's knowledge, helps in effective learning. For this purpose, the learner model captures several dimensions (such as listening, speaking, writing, and reading ability) of what it means to understand each concept in the knowledge map, and allows each assessment interaction a child does to record a normalized update in the model. We identify that different assessment types provide different amount of information gain on the learner's understanding of a word. For instance, single-answer multiple choice questions, multiple-answer multiple choice questions, blank filling, matching activities, etc. have different amount of information gain mainly due to different probabilities of guessing.

3 Entropy-Based Normalization of Learner Model

To achieve normalization across assessment types providing different amounts of information, we develop a common framework based on information theory and entropy [11]. Consider a typical tutoring system [4] having N users trying to learn a domain model of K words. The system updates learner model L_i for user i, for $i = 1, \ldots, N$, using assessments of different types. Let there be M assessments A_j, for $j = 1, \ldots, M$, and each assessment has a corresponding assessment type $t_j \in \{1, \ldots, T\}$ where T is the number of distinct assessment types. Each assessment A_j is also associated with a set of words $\mathcal{C}_j \subset \{1, \ldots, K\}$ that it tests. Following an assessment, we propose the following steps to be undertaken for learner model update:

1. Identify learner id i and assessment A_j.
2. Identify the concept set $\mathcal{C}_j$ associated with the assessment A_j.
3. **Identify the assessment type t_j and the normalization parameter w_{t_j} associated to the assessment type.**
4. For each of the concepts in $\mathcal{C}_j$, update its knowledge score using the **normalized version of** received response to assessment A_j.

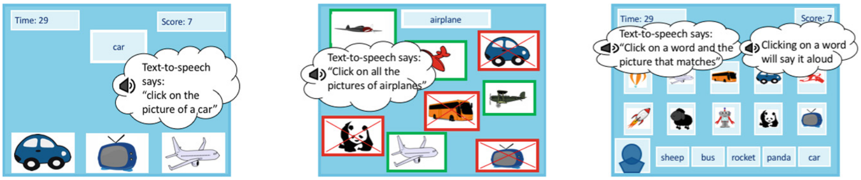
In the typical learner model update, assessment type is ignored and there is no step 3. This step is typically ignored due to a couple of reasons. The tutoring system either consists of only one assessment type or it does not explicitly attempt to incorporate multiple assessment types. If we treat all assessments as same, it results in incorrect learner model updates since different assessment types provide different information. For example, a binary choice assessment question is clearly *easier* than a multiple choice assessment with more than 2 choices.

Here, we introduce an additional parameter that incorporates the assessment type and ensures normalized updates to a learner model. This parameter depends on the *information content* in the assessments of a certain type. Let w_t for $t \in \{1, \ldots, T\}$ be the parameter that quantifies the information content present in the assessments of type t. The additional normalization step ensures the learner model is updated consistently across the different activities/experiences that the learner undergoes during the learning continuum. The normalization parameter w_t is a function of the information available in the assessments.

4 Formative Assessment Prototypes

4.1 Multiple Choice Questions (MCQs)

The most common type of assessments are the MCQs with only one correct answer. Figure 2(a) shows an example MCQ in vocabulary learning. The knowledge of the word *car* is tested by giving multiple images to select, with one correct image and two wrong images. For an MCQ with Q choices, we receive $\log_2 \frac{1}{p_c}$ bits of information per assessment where p_c is the probability of choosing the correct answer. This probability p_c depends on the quality of distractor choices. Considering all choices would be equally probable, a random choice results in $p_c = \frac{1}{Q}$. After choosing an exponential relation between normalization parameter and information, we get $w_t = 1 - \exp(-\log_2 Q)$ for MCQs with Q choices.


Time: 29
Score: 7
car
Text-to-speech says: "click on the picture of a car"
airplane
Text-to-speech says: "Click on all the pictures of airplanes"
Time: 29
Score: 7
Text-to-speech says: "Click on a word and the picture that matches"
Clicking on a word will say it aloud
sheep
bus
rocket
panda
car

(a) (b) (c)

Fig. 2. Different types of assessment examples. (a) MCQ. (b) MRQ. (c) MTF.

4.2 Multiple Response Questions (MRQs)

A variant of MCQs which are typically considered more difficult are the MRQs. MRQs are similar to MCQs as they have single question with multiple choices but more than one correct choice. Figure 2(b) shows an example MRQ in vocabulary learning where all images representing the word *airplane* are to be picked. Clearly any number of the eight choices could represent *airplane*. For an MRQ with Q choices, any of those could be a correct answer. Each such MRQ is equivalent to a series of Q binary choice questions. Let $q = 1, \ldots, Q$, be the index for the choices; the response corresponding to binary choice q of the question is providing us with $\log_2 \frac{1}{p_q}$ bits of information, where p_q is the probability that the correct selection has been made for the choice q. For the case when all choices are of medium quality, then each of them provides $\log_2 \frac{1}{0.5} = 1$ bit of information. Therefore, the learner model is updated by a series of Q binary responses (0/1 based on the individual choice correctness), after weighing them by $w_t = 1 - \exp(-1)$ (using the same exponential function as before). For example, if the learner has given P out of Q correct responses[1] in MRQ, then update the learner model by considering a series of P 1's followed by (Q-P) 0's, weighed by $w_t = 1 - \exp(-1)$.

[1] A choice is wrong if it is the correct answer and is not selected, and vice versa.

4.3 Match the Following

In a typical MTF assessment, R questions are provided along with Q answer choices ($Q \geq R$), with exactly one correct answer per question. The learner matches questions and answers together by selecting question-answer pairs. Figure 2(c) shows an example MTF for matching word-image pairs.

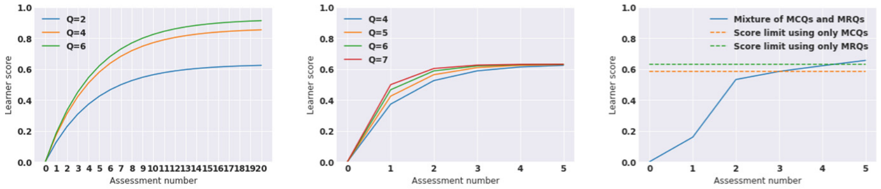
In MTFs, there are Q choices for the first of the R questions and the response has $\log_2 Q$ bits of information. Following the matching of the first question, there are two possible assessment sub-types: where the previously chosen answer choice is replaced with a new choice, thereby resulting in the second question still having Q answer choices, or leaving the set of answer choices unperturbed resulting in $Q-1$ choices for the second question. For each of the MTF sub-type, the learner model is updated in a different manner. For the first sub-type, where all R questions have same number of choices (Q), the learner model is updated by a series of R MCQs with Q choices, and hence each of the updates would use $w_t = 1 - \exp(-\log_2 Q)$. On the other hand, for the MTF sub-type where the answer choices remain unchanged for the entire MTF, the learner model is updated with a series of MCQs with reducing number of choices. In other words, the first of the R questions would have $w_t = 1 - \exp(-\log_2 Q)$, and the r-th question, for $r = 1, \ldots, R$, would have $w_t = 1 - \exp(-\log_2 (Q + 1 - r))$.

4.4 Demonstration Using an Example Learner Model

Consider a learner model where every (learner, word) pair has an associated learning score (between 0 and 1) that determines the confidence that the learner knows the word. The learning score is updated based on incoming assessment data on the (learner, word) pair using the exponentially weighted moving average (EWMA) model given by $l_{n+1} = \alpha l_n + (1-\alpha)x_n$ where l_n is the learning score at time n, α is the EWMA parameter that can be interpreted as the learning rate, and $x_n \in \{0, 1\}$ is the response to the assessment question. The normalization parameter w_t is incorporated into the update equation by re-writing the update equation as $l_{n+1} = \alpha l_n + (1 - \alpha) w_{t_n} x_n$ where w_t is the normalization weight corresponding to the assessment A_n which depends on the type of assessment (t_n). This weight is a function of the amount of information provided by the assessment type in units of bits and is between 0 and 1.

For MCQs, the update rule using w_t becomes $l_{n+1} = \alpha l_n + (1 - \exp(-\log_2 Q))(1 - \alpha)x_n$ where $x_n = 1$ only if the correct response is provided. For MRQs, the update rule using w_t derived in Sect. 4.2 becomes $l_{n+1} = \alpha^Q l_n + (1 - \alpha) \sum_{q=1}^{Q} \alpha^{Q-q}(1 - \exp(-1))x_q$ where x_q, for $q = 1, \ldots, Q$ is the series of Q binary values with P 1's followed by $(Q-P)$ 0's. The above equation simplifies to $l_{n+1} = (1 - \exp(-1))\alpha^{Q-P} - \alpha^Q(1 - \exp(-1) - l_n)$ For MTF, the update rule can be similarly written for the two sub-types using w_t derived in Sect. 4.3. The expressions are similar to the ones derived above depending on the sub-type of MTF.

Figure 3 shows the evolution of the learner score for a learner who just mastered a concept (and hence provides a series of correct responses to MCQs and



(a) (b) (c)

Fig. 3. Learner score evolution. (a) For a series of MCQs. (b) For a series of MRQs. (c) For a mixture of 6 MCQs and MRQs in comparison to the learner score achievable with same number of only MCQs and only MRQs.

MRQs) as per the EWMA update rule using the described normalization.[2] One can observe that MRQs converge faster than MCQs but to a lower score due to the small weights per unit choice in MRQs. Therefore, MCQs and MRQs provide different kind of information. This observation highlights an interesting tradeoff between (1) using the easier variants for assessing learners with more engaging experiences, and (2) using the harder variants (such as blank filling or open-ended questions) to converge to a higher score for better confidence on knowledge of a concept. Figure 3(c) presents the learner score evolution for a mixture of alternate MCQs and MRQs with fixed parameter of Q ($Q = 3$ for MCQ and $Q = 7$ for MRQ). The learner score for the mixture converges to a value higher than when the same number of only MCQs or MRQs are used. The tutoring platform can decide on the choice of assessment types and order of assessments in order to achieve rapid convergence while balancing engagement.

5 Discussion and Conclusion

Table 1 presents an example list of existing popular applications whose assessments are relevant to early-learners. The assessments could be mapped to one or more of the categories. For instance, the right column of the table shows the

Table 1. Existing early-age learning applications

Application	Learning domain	Assessment type
Duolingo	Language	MCQ ($Q = 4$)
Khan academy	Vocabulary and math	MCQ and MRQ
The phrasal verbs machine lite	Verb-preposition	MCQ ($Q = 5$)
Fish school	Alphabets, colors, etc	MTF
Stack the states	USA geography	MCQ ($Q = 4$)

[2] For the simulations, $\alpha = 0.8$ and $l_0 = 0.001$.

assessment types each of the applications cover. Besides the applications discussed in Table 1, there are others that use puzzle and sorting games, but they are not focused towards assessing concepts but instead assess skills. Finally, this work is a step towards enabling open and unified learner models for a variety of experiences young learners choose for learning, and more work is needed to understand the generalization of this framework to other domains beyond vocabulary. By developing theoretically-driven normalization parameters to the learner model update rules, we ensure all assessment information from learner is incorporated in an appropriate manner.

References

1. Biemiller, A.: Vocabulary: what words should we teach. Better Evid. Based Educ. Lang. Arts **3**, 10–11 (2011)
2. Biemiller, A.: Which words are worth teaching? Perspect. Lang. Lit. **41**(3), 9 (2015)
3. Biemiller, A.: Words Worth Teaching: Closing the Vocabulary Gap. McGraw-Hill SRA, Columbus (2010)
4. Brusilovskiy, P.L.: The construction and application of student models in intelligent tutoring systems. J. Comput. Syst. Sci. Int. **32**(1), 70–89 (1994)
5. Desmarais, M.C., Baker, R.S.J.: A review of recent advances in learner and skill modeling in intelligent learning environments. User Model. User-Adap. Inter. **22**(1), 9–38 (2012)
6. Farkas, G., Beron, K.: The detailed age trajectory of oral vocabulary knowledge: differences by class and race. Soc. Sci. Res. **33**(3), 464–497 (2004)
7. Hart, B., Risley, T.R.: The early catastrophe: the 30 million word gap by age 3. Am. Educ. **27**(1), 4–9 (2003)
8. Beck, I.L., McKeown, M.G., Kucan, L.: Bringing Words to Life: Robust Vocabulary Instruction. Guilford Press, New York (2013)
9. Neuman, S.B., Wright, T.S.: All About Words: Increasing Vocabulary in the Common Core Classroom, Prek-2. Teachers College Press, New York (2013)
10. Prentzas, J.: Artificial intelligence methods in early childhood education. Artif. Intell. Evol. Comput. Metaheuristics **427**, 169–199 (2013)
11. Shannon, C.E.: A mathematical theory of communication. Bell Syst. Tech. J. **27**, 379–423 (1948)

Exploring Students' Behaviors in Editing Learning Environment

Xuebai Zhang[2], Xiaolong Liu[1](✉), Shyan-Ming Yuan[2](✉), Chia-Chen Fan[2], and Chuen-Tsai Sun[2]

[1] College of Computer and Information Sciences, Fujian Agriculture and Forestry University, Fuzhou 350002, China
xlliu@fafu.edu.cn

[2] Department of Computer Science, National Chiao Tung University, Hsinchu 300, Taiwan
asuracocoa@gmail.com, {smyuan,ctsun}@cs.nctu.edu.tw, wandy260178@yahoo.com.tw

Abstract. As a pathway for learning, remix has become one of the most important practices within the field of open educational resources. In this study, we investigate the searching and re-editing behavior of students in an online web environment. Participants were asked to search and remix the retrieved web information, which was based on the content of textbook. To explore learning process in the remixing environment, the relationships among function of thinking styles, the search behaviors, the edit behavior, pre-performance, and the final remixing performance are analyzed. Various behavior data are recorded, including web visiting log, interaction log and eye tracking data. The finding provides insight into how to understand the behaviors associated with underlying cognitive and learning performances.

Keywords: Thinking style · Eye tracking · Remix · Learner cognitive

1 Introduction

Remix as the reworking and combination of existing creative artifacts is becoming one of the most important practices within the field of open educational resources [1]. Remixing online is described as important learning, which represents a low-cost and accessible form of participation. Many small contributions are aggregated and remixed toward high quality information goods, and learning happens through the aggregation process as a form of legitimate peripheral participation. The concepts in textbook are sometimes too abstract and complicated to be comprehensible. Whereas, search and retrieve process help students grasp the concept completely and accurately [2]. During the remixing process, students reconstruct the selected information and connect the pieces of knowledge to produce their own storytelling line. With an increasing number of web information, students can potentially download and stream media whatever, wherever and whenever they like, and affording great flexibility in learning experiences [3]. This flexibility combined with the social tools can help to provide opportunities to remix by taking the best from a range of approaches [4].

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 399–404, 2018.
https://doi.org/10.1007/978-3-319-91464-0_46

Markham's study [4] indicated that remix relies on sampling, borrowing, and creatively reassembling to develop something that is used to persuade others. It allows learners to directly edit and fully control the contents of their textbook. In this process, students can conduct a series of cognitive activities such as self-monitoring, and they may feel more confident if the remixed results are validated by teacher. While current researches of remixing have explored methodologies for promoting remixing and theories about the quality of remixed outcomes, we know little research that remix to produce creative work based the content of textbook.

Information-seeking and remixing behaviors are complex cognitive processes. Individuals' differences in the remixing process are reliant on many factors, including intellectual differences, beliefs, judgments, thoughts, emotional trends, attitudes, values, and experience from past. Some of the most important variables include self-efficacy, critical thinking, and thinking styles [5]. Thinking styles, as an individual-difference variable in human performance, have attracted the attention of many scholars and educational psychologists [6]. Different with ability that refers to what one can do, thinking style refers to how one prefers to use one's abilities. As one of the most famous theories of styles, Sternberg's theory of mental self-government [7] describes the function dimension (legislative, executive and judicial styles) of thinking styles. The detail of the function dimension is as follows: (a) Legislative: They prefer the problems which require them to devise, design, and giving commands. In other words, they create their own laws; (b) Executive: Individuals with executive way of thinking prefer to accomplish commands and instructions. They, therefore, like to be guided by others. They would rather deal with administrative jobs and restricted laws; (c) Judicial: The advocators of this style care about judgment and assessment of things. They prefer problems that allow them to analyze and evaluate the attitudes and affairs [8]. Related studies show that cognitive factors such as thinking style are influential in creativity and performance [9, 10], thus in this study we focus on argue that thinking style are the critical factors that influence remix behavior and performance.

In this paper, remix is used as a strategy for thinking about learning design while using a high school research study to inform students' behaviors before and during remixing. We investigate students' searching and editing behaviors when they remix their own textbook after searching and retrieving. Through this remixing process, students are able to reconstruct learned knowledge in cognitive from a perspective of knowledge producer or creators. While thinking styles are considered to have influence to the creating process and performance, we explore the effects of thinking styles on behaviors and performances in this study.

2 Method

Seven participants were recruited from one class of the first grade students in a senior high school located in northern Taiwan. Four of them were female, three were male, and their age ranged from 14 to 15. All of them took Introduction to Computers courses in the past two months and had sectional exam scores. Students had practiced basic computer and web literacy skills and at least knew how to use web browsers and

Microsoft sway. Therefore, they already possessed some prior knowledge and basic skills for searching and re-editing the textbook.

Microsoft Sway is a presentation program that allows students to create a beautiful, interactive, web-based expression of their ideas from browser. It provides a unique storytelling series for participants to type, insert, edit, and format the content. The type of content can be text, images, videos, and even Office documents. Participants are also able to add content from various sources into their Sway presentations, including YouTube, Facebook, Mixcloud and so on. Sway takes care of the design work and helps students focus on the human part: their ideas and how they relate to each other. This contributes to students' engaging in construction. Students can preview their own work at any time to see how it will appear to others during editing and later decide to share their work with peers.

The student's goal task is to rewrite a taught section of Introduction to Computers textbook. When participants began the experiment, they are presented with a blank sway temple and the textbook (.pdf) opened in chrome browser. They are given a total of 30 min to complete their overall task through two sections: searching section and editing section. Participants are first asked to explore the related content by searching webpages within advised 10 min. In this section, they are required to leave seven pages that they think most relevant for the following editing. After searching, students begin to edit their story of the texture through the sway within remain time. They are required to navigate through remain pages and the origin text content.

After task, participants filled out a demographic questionnaire and a thinking styles questionnaire (TSI). The TSI was adopted from Sternberg's Thinking Style Questionnaire—Functions Dimension (legislative, executive, and judicial styles) [11]. It measured students' mental self-government with 15 statements and 3 subtests. Each subtest comprised of 5 questions that evaluate one thinking style. Each question uses a 7-point Likert scale (1-disagree to 7-agree). The Cronbach's α of legislative type was .65; executive type, .82; Judicial type, .89; and the whole scale, .90; these values were regarded as acceptable. The higher the score is, the more the person is inclined to prefer that thinking style. The mean scores of the subscale in our sample were legislative, 23.29 (SD = 5.25); executive, 27.43 (SD = 5.03); and Judicial, 22.57 (SD = 6.53).

The performance of each work submitted by the participants was graded by two teachers and three peers. Teacher's assessment consisted of 7 items with three items asking about layout (Suitable graphic, creative, clear), three items asking about content (complete, accurate and explicit), and one items asking about teaching availability. The items of layout and content assessed by peers and teachers were identical. Peers were also asked about their perception of own preferences to the works. All the items were rated from 1 to 7 points. The score of the complete work of each participant is provided based on the sum of all those items evaluated by teachers and peers, and by an equal split of both.

3 Behavior Data Analysis

When students searched and edited, we collected multi-channel process data, including (1) interaction log files and (2) eye tracking data, using Ogama-plus, an open source package for gaze data and interaction data analysis provided by NCTU DCS-LAB [12].

The log file captures participants' interaction with browser, such as browsing sequence and mouse action type. The eye tracking data was collected using an Eye Tribe Tracker, and take the form of fixations on a single point.

The search phase is defined from entering the first keyword to find the target content; the edit phase is defined from entering the first word in the sway. The following are indicators that quantify learners' search processes and editing process as behaviors: (a) The total number of viewed webpages during the experiment, which reflects the range of the information search; (b) The time spend during visiting each webpage in searching and editing, i.e., the amount of time that participants spent viewing each pages; (c) The number of strategy changes, i.e., the participants changed from viewing a page to viewing another page. We further categorized the strategy changes based on page content as follows: (a) the total number of page changes in searching and editing; (b) the frequency of visiting textbook in both phases; (c) the frequency of visiting the sway page in both phases; (d) Gaze behavior: the fixation number and fixation duration for each visit in searching and editing phases; (e) Mouse interaction logs: the number of left clicks and right clicks for each visit in searching and editing phases.

4 Results

The results of the data pertaining to correlations are shown in Table 1. Significant correlations were found only between the legislative thinking style and the total number of webpage changes during editing ($r = .773$, $p < .05$). Significant correlations were found between the executive thinking style and the right clicks each visit in searching phase ($r = -.820$, $p < .05$), the fixation number each visit during editing ($r = -.823$, $p < .05$), the fixation duration for each visit during editing ($r = -.811$, $p < .05$), the right clicks each visit during editing ($r = -.766$, $p < .01$), and the content score of the editing performance ($r = .761$, $p < .05$). In addition, significant correlations were found between the Judicial thinking style and the time spend visiting each page ($r = -.801$, $p < .05$), the total number of page changes during editing ($r = .791$, $p < .05$), the frequency of visiting textbook in edit phase ($r = .779$, $p < .05$), sectional exam score ($r = .771$, $p < .01$), and the content score of the final editing performance ($r = .869$, $p < .05$).

The correlation results also show that significant negative correlations were found between the numbers of right clicks for each visit during the edit phase and the pre performance of the participants ($r = -.787$, $p < .05$). Significant negative correlations were found between the amount of time for each visit during edit and the content score of the final work score ($r = -.784$, $p < .05$). It also shows significant positive correlations between the sectional exam score and the content of the final work ($r = .789$, $p < .01$).

Table 1. The coefficient of the correlation between behaviors and learner thinking styles.

Behaviors	Thinking styles		
	Legislative	Executive	Judicial
Search behaviors:			
Time spend visiting each page (s)	−.117	.250	−.141
Total # page changes/total search time (count/s)	−.064	−.163	−.056
# view textbook/total search time (count/s)	.069	−.043	.123
Gaze: fixation number each visit	.244	−.529	−.034
Gaze: fixation duration each visit (s)	.191	−.452	−.080
Mouse: left clicks each visit	−.173	.217	−.158
Mouse: right clicks each visit	−.436	−.820*	−.533
Edit behaviors:			
Time spend visiting each page (s)	−.746	−.587	−.801*
Total # page changes/total edit time (count/s)	.773*	.413	.791*
# view textbook/total edit time (count/s)	.753	.608	.779*
# visit sway/total edit time (count/s)	.180	−.142	.151
Gaze: fixation number each visit	.139	−.823*	−.122
Gaze: fixation duration each visit (s)	.134	−.811*	−.141
Mouse: left clicks each visit	−.636	−.332	−.537
Mouse: right clicks each visit	−.191	−.766*	−.429
Pre performance:			
Sectional exam score	.556	.714	.771*
Work performance:			
Layout score	.157	−.095	.248
Content score	.688	.761*	.869*
Total score	.400	.453	.610
Other:			
Total # viewed pages	.266	−.352	.299

5 Conclusion

This study explored how human thinking styles are associated with search behaviors, edit behaviors, pre performance, and re-editing performance in online editing learning environment. Our finding indicated that during search phase, executive style is related to only one indicator, right clicks for each visit. Students with better executive ability would use right clicks less during searching. As students use right clicks to download the retrieve pictures or document, less right clicks indicate less download actions. During edit phase, executive style is negative related with the both fixation indicators and right clicks for each visit. Students with better executive skills tended to take less time and less number gaze on each page, and less right clicks number. Legislative style is positive associated with total number of page changes, which indicate that student with better legislative skill tended to change the pages more frequency. Judicial style is negative associated with time spend visiting each page. However, it is positive related with total number of page changes and number viewing textbook. Students with better judicial skill also tended to score high in pre-performance and content performance in remixing work. In the future, more discussion will be given to understand students' behavior during remixing editing.

Acknowledgement. The work was supported by the fund of MOST of Taiwan (Grant No. 105-2511-S-009-007-MY3 and No. 105-2511-S-009-004-MY3), MOE (Ministry of Education in China) Project of Humanities and Social Sciences (Grant No. 17YJC880076), Educational Commission of Fujian Province, China (No. JAS170171).

References

1. Amiel, T.: Identifying barriers to the remix of translated Open Educational Resources. Int. Rev. Res. Open Distance Learn. **14**(1), 126–144 (2013)
2. Lei, P.L., Sun, C.T., Lin, S.S.J.: Effect of metacognitive strategies and verbal-imagery cognitive style on biology-based video search and learning performance. Comput. Educ. **87** (C), 326–339 (2015)
3. Thomson, A., Bridgstock, R., Willems, C.: "Teachers flipping out" beyond the online lecture: maximising the educational potential of video. J. Learn. Des. **7**(3), 67–78 (2014)
4. Markham, A.: Remix cultures, remix methods: reframing qualitative inquiry for social media contexts. In: Denzin, N., Giardina, M. (eds.) Global Dimensions of Qualitative Inquiry, pp. 63–81. Left Coast Press, Walnut Creek (2013)
5. Ashoori, J.: Relationship between academic achievement and self-efficacy, critical thinking, thinking styles and emotional intelligence in nursing students. Sci. J. Hamadan Nurs. Midwifery Fac. **22**(3), 15–23 (2014)
6. Zhang, L.F.: Thinking styles and cognitive development. J. Genet. Psychol. **163**(2), 179–195 (2002)
7. Sternberg, R.J.: Thinking Styles. Cambridge University Press, New York (1997)
8. Abolghasem, P., Dehghankar, L., Jafarisani, M., Badiee, S.E., Tatari, M., Khalafi, A.: Studying the association between thinking styles and creativity among students. Nova J. Humanit. Soc. Sci. **5**(2), 1–7 (2016)
9. Emamipour, S., Seif, A.A.: Developmental study of thinking styles in students and its relation to creativity and academic achievement. J. Educ. Innov. **2**, 3 (2003)
10. Kadivar, P., Javadi, M.J., Sajedian, F.: The relationship of thinking styles and self-regulation with achievement motivation. J. Psychol. Stud. **2**(6), 34 (2010)
11. Sternberg, R.J., Grigorenko, E.L.: Styles of thinking in the school. Eur. J. High Ability **6**, 201–219 (1995)
12. NCTU-DCS LAB. http://dcslab.nctu.edu.tw/. Accessed 21 Nov 2017

Posters

Analysing Problem Sequencing Strategies Based on Revised Bloom's Taxonomy Using Deep Knowledge Tracing

Sweety Agrawal(✉) and Amar Lalwani

funtoot, Bangalore, India
{sweety.agrawal,amar.lalwani}@funtoot.com

Abstract. Revised Bloom's Taxonomy (RBT) is hierarchical in nature and it serves as a common vocabulary for the teachers to classify learning objectives of a curriculum. In this work, we study the effects of using RBT as a problem sequencing strategy on students' learning. We compare a *blocking* strategy based on RBT against the random strategy. We also implement the reversed hierarchical order of this taxonomy as a strategy to understand the effect of a contrast behaviour, if any. We also examine both forward and reverse hierarchical orders by enhancing them with *interleaving* behaviour. We use deep learning based knowledge tracing model, Deep Knowledge Tracing to simulate the students' behaviour. We observe that forward hierarchical order yields a significant gain over reverse hierarchical order. Interestingly, *interleaving* on RBT did not outperform *blocking* strategy as expected [6].

Keywords: Deep knowledge tracing · Revised bloom's taxonomy
Problem sequencing · Intelligent tutoring systems · Interleaving
Blocking

1 Introduction

A Revised Bloom's Taxonomy (RBT) proposed in [1] is hierarchical in nature like the original Bloom's Taxonomy. The six major categories in RBT are - Remember, Understand, Apply, Analyse, Evaluate and Create.

One difference between the two taxonomies is that the last two categories are reversed in the RBT. Here too, like original taxonomy, the categories differ in their complexities, Remember is less complex than Understand, Understand is less complex than Apply, and so forth. Another difference in the Revised Taxonomy is that the complexity of the six categories are allowed to overlap. This relaxes the strict hierarchical assumption. But, nonetheless, the categories do form a hierarchy [1, Appendix A].

We started the proposed study in a quest to jot down a strategy which will enable us to lead our students to mastery. All students can achieve expertise in a domain if two conditions are met [2]: (1) domain knowledge is appropriately

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 407–410, 2018.
https://doi.org/10.1007/978-3-319-91464-0

analysed into a hierarchy of skills and (2) learning experience is structured to ensure that students master prerequisite skills before moving on to higher order skills in the hierarchy.

We would like to study the effects generated on student's knowledge acquisition after training students in lower levels first and then moving on to a higher level in the RBT. This type of strategy is termed as a *blocking* strategy, where a student practices a skill in a block and then moves on to other skills.

Studies [6] have shown that students learn better when they are given repeated exposure to different skills in an *interleaved* manner rather than *blocking*. In the proposed study, we also simulate interleaving strategies on the thinking levels of RBT and compare it with random strategy. This method of interleaving helps us examine how necessary the second condition above is for students to achieve expertise.

We also reverse the order of the RBT, and move students from higher order to the lower order and see the effects of providing students with difficult questions first and whether or not it improves their chances of solving lower level questions.

In funtoot [3], an adaptive learning platform, a sub-sub-concept (*ssc*) is a smallest teachable unit. And problems are available for students to work on in the *sscs*. Every problem in each *ssc* is mapped with the level in the RBT based on the cognitive skill a problem requires. We call this tagging as *btlo* - Bloom's Taxonomy Learning Objective.

We have used a recurrent neural network based knowledge tracing model called Deep Knowledge Tracing [5] to simulate the students' behaviour in all the problem sequencing strategies that we are interested in studying.

2 Dataset and Experiments

The dataset used in this study consists of 41.7 million data points involving 1,03,593 students and 10,158 problems in 536 *sscs* (having at least 2 *btlos*). A data point here represents the interaction between the student and the given problem.

We have trained a Deep Knowledge Tracing (DKT) [5] model on this dataset for each *ssc* and they are based on their respective *btlos* (used as features). The average AUC of the DKT models is 0.71 ($\sigma = 0.06$). We use this DKT model as a virtual student to get the student responses for all the strategies in an *ssc*.

Each strategy delivers a total of 10 problems per *btlo*. Consider an *ssc* *s* having three *btlos*: Remember (R), Understand (U) and Apply (Ap). Table 1 shows all the problem sequencing strategies that were simulated in an *ssc* for this study.

Table 1. Strategies

Strategy	Description
Random	a *btlo* is randomly sampled for which a problem is presented, sequence in s: $U_1 - R_1 - Ap_1 - Ap_2 - U_2 - R_2 - R_3 - ...$
Interleaving-1	One problem is given to a student from each *btlo* in the order of the hierarchy, sequence in s: $R_1 - U_1 - Ap_1 - R_2 - U_2 - Ap_2 - R_3 - ...$
Interleaving-2	Two consecutive problems are given to a student from each *btlo* in the order of the hierarchy, sequence in s: $R_1 - R_2 - U_1 - U_2 - Ap_1 - Ap_2 - R_3 - R_4 - U_3 - ...$
Interleaving-5	Five consecutive problems are given to a student from each *btlo* in the order of the hierarchy, sequence in s: $R_1 - ... - R_5 - U_1 - ... - U_5 - Ap_1 - ... - Ap_5 - R_6 - ... - R_{10} - U_6 - ...$
Forward Blocking	Ten consecutive problems are given to a student from each *btlo* in the order of the hierarchy, sequence in s: $R_1 - R_2 - ... - R_{10} - U_1 - U_2 - ... - U_{10} - Ap_1 - Ap_2 - ... - Ap_{10}$
Reverse Interleaving-1	One problem is given to a student from each *btlo* in the reversed order of the hierarchy, sequence in s: $Ap_1 - U_1 - R_1 - Ap_2 - U_2 - R_2 - Ap_3 - ...$
Reverse Interleaving-2	Two consecutive problems are given to a student from each *btlo* in the reversed order of the hierarchy, sequence in s: $Ap_1 - Ap_2 - U_1 - U_2 - R_1 - R_2 - Ap_3 - Ap_4 - U_3 - ...$
Reverse Interleaving-5	Five consecutive problems are given to a student from each *btlo* in the reversed order of the hierarchy, sequence in s: $Ap_1 - ... - Ap_5 - U_1 - ... - U_5 - R_1 - ... - R_5 - Ap_6 - ... - Ap_{10} - U_6 - ...$
Reverse Blocking	Ten consecutive problems are given to a student from each *btlo* in the reversed order of the hierarchy, sequence in s would be like: $Ap_1 - Ap_2 - ... - Ap_{10} - U_1 - U_2 - ... - U_{10} - R_1 - R_2 - ... - R_{10}$

3 Results and Conclusion

For a *btlo*, the average gain achieved by a strategy can be computed by taking the difference of final and initial probabilities of that *btlo*.

Based on the average gains, we observed that forward blocking is much better than reverse blocking especially in the higher order *btlos*. Forward blocking performed better than forward interleaving strategies. Forward interleaving and reverse interleaving gave similar gains. Hence, in conclusion, interleaving as such did not outperform blocking as seen in the previous work [6].

This results are puzzling since our work in [4] has shown that Revised Bloom's Taxonomy does not have a strict prerequisite structure especially in the higher order levels. Moreover, the findings indicate significant overlap even across non-adjacent levels. We need to study this further to investigate why forward blocking performs well even though there is no strict hierarchical structure.

In this work, we have implemented interleaving on the levels of the hierarchy and we did not see any advantage of it over blocking. But in literature, the interleaving technique is applied on the skills (not necessarily following the prerequisite structure). We need to study this in detail if this is the reason behind interleaving not performing better than blocking.

References

1. Anderson, L.W., Krathwohl, D.R., Airasian, P., Cruikshank, K., Mayer, R., Pintrich, P., Raths, J., Wittrock, M.: A Taxonomy for Learning, Teaching and Assessing: A Revision of Blooms Taxonomy, vol. 9, issue 2, pp. 137–175 (2001)
2. Corbett, A.T., Anderson, J.R.: Knowledge tracing: modeling the acquisition of procedural knowledge. User Model. User-Adap. Iinter. **4**(4), 253–278 (1994)
3. Lalwani, A., Agrawal, S.: Few hundred parameters outperform few hundred thousand? In: Educational Data Mining (2017)
4. Lalwani, A., Agrawal, S.: Validating revised bloom's taxonomy using deep knowledge tracing. In: International Conference on Artificial Intelligence in Education (2018, to appear)
5. Piech, C., Bassen, J., Huang, J., Ganguli, S., Sahami, M., Guibas, L.J., Sohl-Dickstein, J.: Deep knowledge tracing. In: Advances in Neural Information Processing Systems. pp. 505–513 (2015)
6. Rohrer, D., Dedrick, R.F., Stershic, S.: Interleaved practice improves mathematics learning. J. Educ. Psychol. **107**(3), 900 (2015)

MetaMentor: A System Designed to Study, Teach, Train, and Foster Self-regulated Learning for Students and Experts Using Their Multimodal Data Visualizations

Roger Azevedo[1](✉), Nicholas V. Mudrick[1], Michelle Taub[1],
James Lester[2], Robert Taylor[2], Robert Sawyer[2], Kirby Culbertson[2],
and Candice Roberts[3]

[1] Department of Psychology, North Carolina State University,
Raleigh, NC, USA
{razeved,nvmudric,mtaub}@ncsu.edu
[2] Department of Computer Science, North Carolina State University,
Raleigh, NC, USA
{lester,rgtaylor,rssawyer,ksculbe2}@ncsu.edu
[3] Natural Sciences Department, Wake Technical Community College,
Raleigh, NC, USA
cmroberts4@waketech.edu

Abstract. MetaMentor is an interactive system designed to study, teach, train, and foster self-regulated learning (SRL) for students and domain experts using their multimodal data visualizations while they solve complex science problems using multimedia materials. The system is based on contemporary theories of SRL [1], research on human and computerized tutoring [2, 3], and emerging interdisciplinary research on the use of multimodal data ([4] e.g., log files, eye tracking, screen recordings, concurrent verbalizations, facial expressions of emotions, physiological sensors) used to detect, track, model, and foster cognitive, affective, metacognitive, and motivational (CAMM) SRL processes *during* learning and problem solving with advanced learning technologies (ALTs) such as intelligent tutoring systems (ITSs).

Advances in intelligent systems require extending current models and theories by focusing on both students' and experts' multimodal CAMM SRL process data during learning and problem solving, and instructional decision making. Despite the mounting evidence regarding the importance of CAMM SRL processes for understanding human learning and designing intelligent systems, there is no framework, model, or empirical data on the (1) quantitative and qualitative changes in students' multimodal data based on experts' real-time tutoring interventions and their impact on students' developing SRL competencies and domain knowledge; (2) the effectiveness of providing experts with students' real-time multimodal SRL data to augment their instructional decision-making on (1); and (3) experts' multimodal data during tutoring interactions to (a) develop a model of experts' CAMM SRL processes to (b) understand how they self-regulate and how their monitoring and regulatory processes influences their

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 411–414, 2018.
https://doi.org/10.1007/978-3-319-91464-0

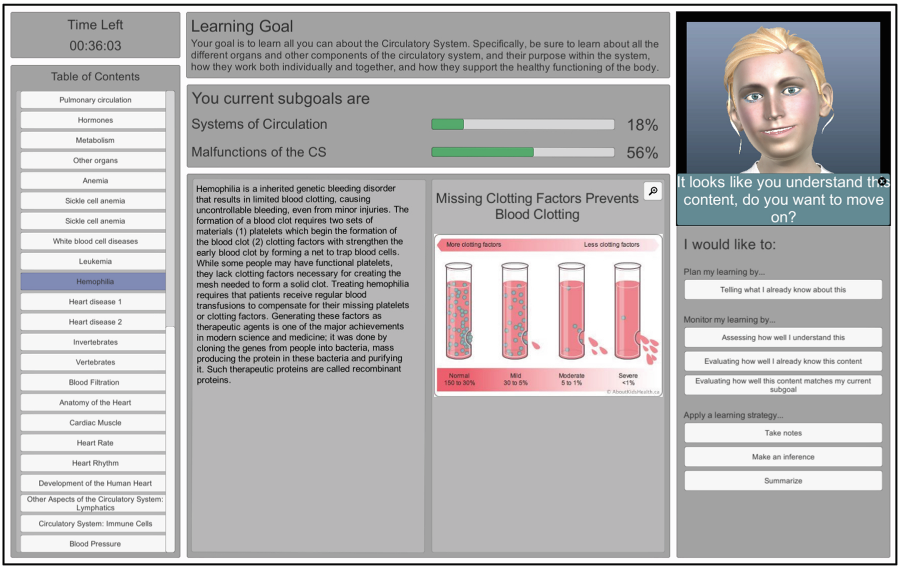
understanding of students' CAMM SRL processes and domain knowledge, which ultimately influences their external regulatory processes. As such, this prototype system will be tested extensively to collect rich temporally unfolding CAMM SRL multimodal student and tutor data to facilitate the creation of an ITS capable of providing adaptive real-time support for students and tutors.

The ultimate goal of MetaMentor is for it to become an intelligent system that is capable of real-time interaction or prerecorded playback of previously collected multimodal data and have tutors practice, train, and externally regulate the student's self-regulation and problem solving across complex science topics and ALTs. For example, an intelligent version of MetaMentor could train a novice tutor or pre-service teacher to externally regulate students' negative emotions using cognitive reappraisal by perceptually cueing students' persistent facial expressions of frustration, highlighting erratic gaze behavior indicating a lack of attention to the relevant instructional material, and displaying coded concurrent verbalizations revealing a lack of metacognitive accuracy and how these three types of multimodal student data can be externally regulated by using a script to downregulate the negative emotions via cognitive reappraisal.

In this interactive session, we will do a live demonstration of the system with two instrumented team members (one playing the student and the other playing the tutor) simulating a tutoring session and illustrating their multimodal data. Our focus will be on: (1) describing the architecture of the system, (2) presenting the analytical approach to detecting and modeling multimodal data from the student and tutor, and (3) describing how inferences made by the tutor based on (2) translate into real-time external regulation and instructional decision making designed to foster student's SRL and domain knowledge. We illustrate a typical walkthrough with two figures below.

Figure 1 illustrates the students' interface as he is learning about the human circulatory system and has access to a timer and a learning goal, intelligent virtual human (IVH) that facially expresses different emotions (e.g., confusion. joy, etc., controlled by the human tutor) in response to students' CAMM SRL processes, SRL palette where a student can indicate which cognitive strategies and metacognitive processes they are enacting (they can also verbalize these intentions), multimedia science content, and a table of contents related to several body systems.

In contrast, Fig. 2 illustrates the tutor's interface by showing five key interface elements including (a) student's real-time multimodal data including behavioral actions (e.g., mouse movement) and gaze behavior (green dot) from the eye tracker (top-left), (b) live video stream of student's facial expressions (to detect and infer emotional states; top-right), (c) list of metacognitive and cognitive processes the tutor can click on as they are detected (and inferred) from the student's multimodal data and also show the tutor's gaze behavior (bottom-left), (d) emote codes that allow the tutor to send contextually and instructionally-appropriate facial expression(s) embodied in the IVH, and (e) a text box that provides the tutor with all actions enacted by the learner with chat box that allows that tutor to type and send messages to the student that embody external regulatory moves (e.g., prompt the activation of relevant prior knowledge, provide feedback on the accuracy of metacognitive monitoring, model strategy use, induce positive emotions, and enhance task value and interest) and domain knowledge (e.g., declarative and conceptual) (bottom-right).



Learning Goal

Your goal is to learn all you can about the Circulatory System. Specifically, be sure to learn about all the different organs and other components of the circulatory system, and their purpose within the system, how they work both individually and together, and how they support the healthy functioning of the body.

You current subgoals are

Systems of Circulation 18%

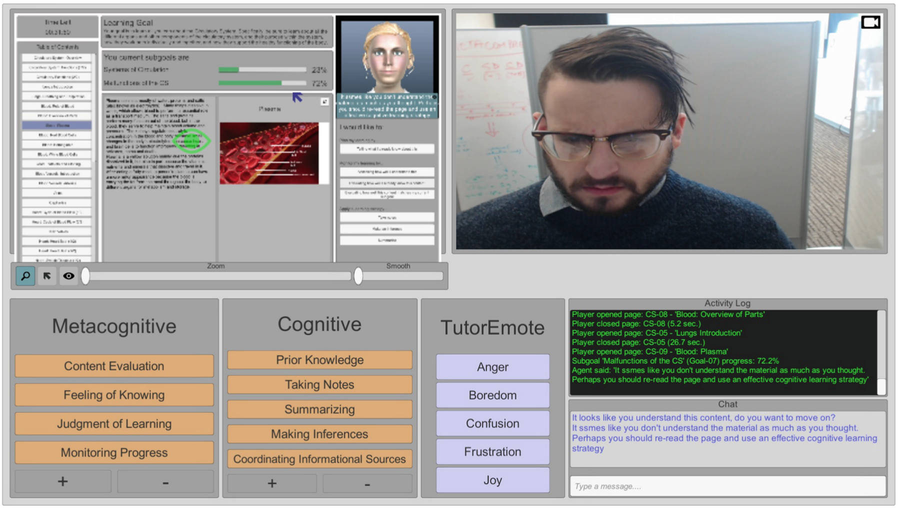
Malfunctions of the CS 56%

Hemophilia is a inherited genetic bleeding disorder that results in limited blood clotting, causing uncontrollable bleeding, even from minor injuries. The formation of a blood clot requires two sets of materials (1) platelets which begin the formation of the blood clot (2) clotting factors with strengthen the early blood clot by forming a net to trap blood cells. While some people may have functional platelets, they lack clotting factors necessary for creating the mesh needed to form a solid clot. Treating hemophilia requires that patients receive regular blood transfusions to compensate for their missing platelets or clotting factors. Generating these factors as therapeutic agents is one of the major achievements in modern science and medicine; it was done by cloning the genes from people into bacteria, mass producing the protein in these bacteria and purifying it. Such therapeutic proteins are called recombinant proteins.

It looks like you understand this content, do you want to move on?

I would like to:
Plan my learning by...
Telling what I already know about this
Monitor my learning by...
Assessing how well I understand this
Evaluating how well I already know this content
Evaluating how well this content matches my current subgoal
Apply a learning strategy...
Take notes
Make an inference
Summarize

Fig. 1. MetaMentor's main student interface.



Activity Log

```
Player opened page: CS-08 - 'Blood: Overview of Parts'
Player closed page: CS-08 (5.2 sec.)
Player opened page: CS-05 - 'Lungs Introduction'
Player closed page: CS-05 (26.7 sec.)
Player opened page: CS-09 - 'Blood: Plasma'
Subgoal 'Malfunctions of the CS' (Goal-07) progress: 72.2%
Agent said: 'It ssmes like you don't understand the material as much as you thought.
Perhaps you should re-read the page and use an effective cognitive learning strategy'
```

Chat

It looks like you understand this content, do you want to move on?
It ssmes like you don't understand the material as much as you thought.
Perhaps you should re-read the page and use an effective cognitive learning strategy

Type a message....

Fig. 2. MetaMentor's main tutor interface.

References

1. Winne, P.H.: Cognition and metacognition with self-regulated learning. In: Schunk, D.H., Greene, J.A. (eds.) Handbook of Self-regulation of Learning and Performance, 2nd edn, pp. 36–48. Routledge, New York (2018)
2. Graesser, A.: Conversations with AutoTutor help students learn. Int. J. Artif. Intell. Educ. **26**, 124–132 (2016)
3. Johnson, W.L., Lester, J.C.: Face-to-face interaction with pedagogical agents, twenty years later. Int. J. Artif. Intell. Educ. **26**, 25–36 (2016)
4. Azevedo, R., Taub, M., Mudrick, N.V.: Using multi-channel trace data to infer and foster self-regulated learning between humans and advanced learning technologies. In: Schunk, D., Greene, J.A. (eds.) Handbook of self-regulation of learning and performance, 2nd edn, pp. 254–270. Routledge, New York (2018)

What Can Eye Movement Patterns Reveal About Learners' Performance?

Asma Ben Khedher(✉), Imène Jraidi, and Claude Frasson

University of Montreal, Montreal, QC, Canada
{benkheda, jraidii, frasson}@iro.umontreal.ca

Abstract. In this paper, we are particularly interested in analyzing learners' visual behaviour and what can fixation-based metrics can reveal about students' learning performance while solving medical cases. The objective of this study is to analyze how the students visually explore the learning environment across different areas of interest. Results showed that even so there is a specific area of interest that has the greatest level of attention from the students, this area does not impact students' performance in the resolution of the clinical tasks. The findings demonstrated that there are other areas of interest that are positively correlated with the learners' success.

Keywords: Eye movements · Students' performance · Serious game

1 Introduction

The use of sensing technologies (e.g. facial expression, galvanic skin response, EEG, eye tracking, etc.) has flourished in the past few years [1–3]. They have proven their efficiency in human-computer interaction systems, especially in educational environments. Due to its ease of use, eye tracking technique has been a very useful tool to track users' eye movements and assess their visual activities, in an effort to understand how students evolve through the learning process using different eye gaze metrics such as saccade length and number of revisits [4–7]. Assessing students' learning outcomes has gained much interest in this last decade. In fact, computer-based adaptive learning environments seek constantly to provide students with adequate help strategies in an effort to foster their learning progress.

In this paper, we propose to use eye tracking to analyze students' eye movements and particularly where their visual attention is focused as they were interacting with a medical serious game. We computed two fixation-based metrics namely; fixation duration (F.D) and time to first fixation (T.T.F.F) to assess students' visual behaviour across different areas of interest and check whether there were particular areas that contributed to the students' success.

2 Method and Material

An experimental study was conducted where eye movements of 15 undergraduate medicine students (7 females) aged between 20 and 27 ($M = 21.8 \pm 2.73$) were

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 415–417, 2018.
https://doi.org/10.1007/978-3-319-91464-0

recorded using a Tobii Tx300 eye tracker as they interacted with a medical serious game. After the calibration process, the game was displayed with a brief introductory scene recalling the main objectives of the game and the tasks they need to fulfil. Participants were invited to interact with the learning environment called Amnesia during 30–45 min.

Amnesia is a medical serious game developed to assess the cognitive abilities of novice students through clinical problem-solving tasks that were validated by a medical professional. The game features a virtual hospital where the players need to prove that they do not suffer from amnesia by resolving six medical cases. For each case, students are instructed to identify the correct diagnosis and the appropriate treatment. They were given three attempts to find out the correct responses.

In order to obtain a detailed analysis of the students' eye movements, specific areas of interest (AOIs) representing task-relevant regions of the screen were created in each medical case. Six AOIs were defined as follows: Information (*I*), Antecedents (*A*), Symptoms (*S*) Analysis (*N*), Diagnosis (*D*) and Treatment (*T*). We were as well interested by performing case-by-case analyses since the medical cases were different in terms of their content.

3 Results and Discussions

Statistical comparison between the different areas of interest using mean values (M) and standard deviations (SD) was performed. The results showed that the Symptoms' area was by far the most fixated zone by all the participants in all cases in terms of fixation duration. The most prominent F.D was computed in the last case (M = 28.72, SD = 17.48). In accordance with these findings, a one-way ANOVA was conducted to investigate whether there were significant differences among all the AOIs in terms of F.D. Statistically significant results ($p < 0.001$) were found among all cases. Tukey post-hoc tests were carried out where all areas of interest were compared by pairs in order to identify which area of interest caught the most the students' attention. The results indicated that the Symptoms' AOI differed significantly ($p < 0.05$) from the other areas in almost all cases. Based on fixation duration, the time dedicated to looking at the Symptoms' area far exceeded the time dedicated to the other areas ($p < .001$).

Two case-by-case MANOVAs were conducted in terms of fixation duration and time to first fixation. In the first one, we focus on the symptoms' area. Results showed no statistically significant relationship between the fixation duration and the identification of the correct diagnosis in almost all cases. In the second MANOVA, we considered all the remaining areas of interest to check whether there is such area that does contribute to the students' success. Case 1 was discarded since all participants succeeded in identifying the diagnosis. Case 2 ($F(1,8) = 4.946$) and case 5 ($F(1,3) = 0.084$) showed no significant differences (p = n.s) between the areas of interest in terms of both fixation metrics. For the remaining three cases the results were statistically significant in terms of time to first fixation. Post hoc tests were analyzed for separate correlational analyses to show which AOI could potentially contribute to the students' success. Results showed that there was not a unique AOI that was associated with participants' performance, but in each case, there were different areas. For case 3 and 6, a significant effect was found for all areas ($p < 0.05$). In case 4, a unique significant result was found for the Antecedents' region ($p < 0.001$) which is actually prominent since this area contains clues that the students have to focus on to identify the correct diagnosis.

In summary, we were able to identify a relationship between some areas of interest and students' outcomes using fixation-based metrics. These findings showed that the symptoms' area of interest caught the most the students' interest, however, it did not contribute to their success. In fact, longer fixations are note always indicators of learning success. For the remaining regions, case-by-cases analyses revealed that different visual activities were found among the learners depending on the medical cases solved. Our main objective in this research was to analyze the students' visual attention through their learning experience and what can eye movements reveal about learners' performance. As future woks, we aim to integrate physiological variables with the eye tracking technique in order to assess students' engagement as well.

References

1. Jraidi, I., Chaouachi, M., Frasson, C.: A dynamic multimodal approach for assessing learners' interaction experience. In: Proceedings of the International Conference on Multimodal Interaction, pp. 271–278 (2013)
2. Jraidi, I., Frasson, C.: Student's uncertainty modeling through a multimodal sensor-based approach. J. Educ. Technol. Soc. **16**(1), 219–230 (2013)
3. Chaouachi, M., Jraidi, I., Frasson, C.: MENTOR: a physiologically controlled tutoring system. In: Ricci, F., Bontcheva, K., Conlan, O., Lawless, S. (eds.) UMAP 2015. LNCS, vol. 9146, pp. 56–67. Springer, Cham (2015)
4. Lallé, S., Conati, C., Carenini, G.: Impact of individual differences on user experience with a visualization interface for public engagement. In: Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization, pp. 247–252 (2017)
5. Raptis, G.E., Katsini, C., Belk, M., Fidas, C., Samaras, G., Avouris, N.: Using eye gaze data and visual activities to infer human cognitive styles: method and feasibility studies. In: Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization, pp. 164–173 (2017)
6. Ben Khedher, A., Jraidi, I., Frasson, C.: Assessing learners' reasoning using eye tracking and a sequence alignment method. In: Huang, D.S., Jo, K.H., Figueroa-García, J. (eds.) ICIC 2017. LNCS, vol. 10362, pp. 47–57. Springer, Cham (2017)
7. Ben Khedher, A., Jraidi, I., Frasson, C.: Local Sequence alignment for scan path similarity assessment. Int. J. Inf. Educ. Technol. **8**(7), 482–490 (2018, in press)

“Hypocrates”: Virtual Reality and Emotions Analysis Towards a Personalized Learning

Marwa Boukadida, Hamdi Ben Abdessalem(✉), and Claude Frasson

Département d’Informatique et de Recherche Opérationnelle, Université de
Montréal, Montréal H3C 3J7, Canada
{boukadim, benabdeh, frasson}@iro.umontreal.ca

1 Introduction

Physiological measurements such as brain activity (EEG) [1], electrodermal skin activity (EDA) [2], and eye tracking [3] helps understand and assess the physiological process of emotions. The use of virtual reality (VR) provide interactive systems that offer to the user a high sense of presence and immersion in the virtual world [4]. Our goal is to analyze the behavior and reactions of medical students in clinical reasoning situations through a VR environment and emotional measures. We propose to follow in real time the emotional state of medical students and to intervene in the virtual environment in order to provoke the frustration or stress and analyze the impact on their performance.

2 Method

2.1 “Hypocrates” System

The implementation of our system is made with the Unity 3D game engine. It contains three main modules: a virtual reality environment, a manager and an intelligent agent. Our goal is to track in real-time the emotional state of medical students and intervene in the virtual environment in order to change their emotional state and analyze subsequently their reactions after mistakes in clinical reasoning (Fig. 1).

Virtual Reality Module. In this environment, the medical student is immersed in several scenes. He initially goes through an introductory scene in which we expose and explain how to interact with this virtual reality environment. Subsequently, he is exposed to a virtual operating room or a doctor’s office, depending on the type of the medical case to solve.

Manager Module. This component uses medical cases and additional medical data to **generate** a problem case which is submitted to the student through the VR environment. The goal is to produce a case with correct and wrong data so that the student will make mistakes if he does not select the correct data. Figure 2 shows such a problem case.

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 418–420, 2018.
https://doi.org/10.1007/978-3-319-91464-0

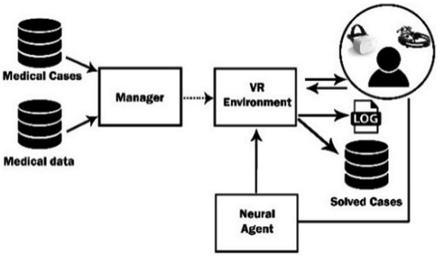
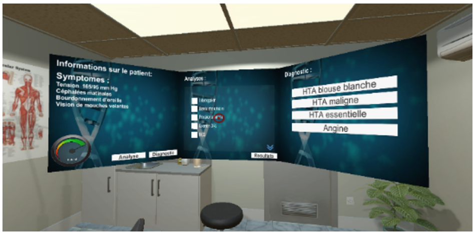


Fig. 1. "Hypocrates" architecture


Informations sur le patient:

Symptomes :
Tension 165/95 mm Hg
Céphalées matinales
Bourdonnement d'oreille
Vision de mouches volantes

Analyses :

Diagnostic :
HTA blouse blanche
HTA maligne
HTA essentielle
Angine

Fig. 2. Example of problem case

Neural Agent. The neural agent is an intelligent agent that receives the different physiological measurements from the measurement tools (in our case, we will use the EEG), uses a rules base to intervene on the virtual environment in order to change the emotional state of the user [5]. In this work, the neural agent runs in real-time and intervenes in the virtual environment in order to create stressful situations and provoke the emotional state of the student in order to see the impact on his decisions.

2.2 "Hypocrates" Functionalities

The Manager extracts medical cases from the database, adds extra medical data and sends it to the VR environment. Meanwhile, the neural agent is running for EEG data capture and intervention in the virtual environment. While the student is interacting with the virtual reality environment, EEG data are collected and saved every second in a time-stamped log file. We also save each decision made by the participant, its type, whether it is correct or incorrect, and the time at which it was made.

3 Experiment and Results

We conducted experiments involving 15 medical students in order to test their reliability using "Hypocrates". The medical student has to read the displayed symptoms, then he asks for analysis if needed (a panel containing a list of analysis will appear) and the results of demanded analysis will appear. Next, he selects a medical diagnosis.

Once the choice of diagnosis is selected a series of panels, each one containing three actions, (one correct two false), appear one by one. The number of these panels depend on the number of actions to perform in the current medical case.

After the experiments, we conducted a paired-samples t-test to compare the frustration of the medical student before and after the mistake. Results show that the average frustration after the mistake compared to the average frustration before the mistake went from 0.441 to 0.551, $\mathrm{t}\,(179) = 11.0075$ and $\mathrm{p} = 0.000 * <0.01$. This result is significant and we can confirm that the average frustration state of medical students after the error is greater than the average frustration before the error of 11%. We note also that participants made 161 correct choices. Results show that the average frustration after the correct choice compared to the average frustration before the correct choice went from 0.519 to 0.463, $\mathrm{t}\,(160) = 7.3272$ and $\mathrm{p} = 0.000 * < 0.01$. This result is significant and allows us to confirm that the average frustration of medical students decreases by 5% after correct action. The neural agent intervenes in the virtual reality environment to provoke the medical student, so we compared the wrong decisions of the participants before and after these interventions. Results show that, after the intervention of the agent and the increase in the level of frustration, the average of the successive wrong increased. These results show that the performance of medical students can decrease with the increase of frustration.

4 Conclusion

In this paper, we proposed a system that allows us to analyze the behavior and the emotional state of medical students while solving medical cases in a virtual reality environment. The results prove that it is possible to generate emotional situations capable of testing decision-making abilities. Future work will aim to learn the behavior and reactions of medical students, in order to predict their actions and warn them when needed, using machine learning techniques and thus, personalized learning.

Acknowledgment. We acknowledge NSERC-CRD and Beam Me UP for funding this work.

References

1. Soleymani, M., Asghari-Esfeden, S., Pantic, M., Fu, Y.: Continuous emotion detection using EEG signals and facial expressions, July 2014
2. Boucsein, W.: Electrodermal Activity. Springer, New York (2012)
3. Ben Khedher, A., Frasson, C.: Predicting user learning performance from eye movements during interaction with a serious game. In: EdMedia: World Conference on Educational Media and Technology. AACE, pp. 1504–1511 (2016)
4. Bohil, C.J., Alicea, B., Biocca, F.A.: Virtual reality in neuroscience research and therapy. Nat. Rev. Neurosci. (2011)
5. Ben Abdessalem, H., Frasson, C.: Real-time brain assessment for adaptive virtual reality game: a neurofeedback approach. In: Frasson, C., Kostopoulos, G. (eds.) BFAL 2017. LNCS, vol. 10512, pp. 133–143. Springer, Cham (2017)

Embedding Speech Technology into Intelligent Tutoring Systems Using the CloudCAST Speech Technology Platform

André Coy[1(✉)], Phil Green[2], Stuart Cunningham[2], Heidi Christensen[2], José Joaquín Atria[3], Frank Rudzicz[4,5], Massimiliano Malavasi[6], and Lorenzo Desideri[6]

[1] University of the West Indies, Kingston, Jamaica
andre.coy02@uwimona.edu.jm
[2] University of Sheffield, Sheffield, UK
{p.green,s.cunningham,heidi.christensen}@sheffield.ac.uk
[3] CV-Library Ltd, Hampshire, UK
jjatria@gmail.com
[4] Toronto Rehabilitation Institute, Toronto, Canada
[5] University of Toronto, Toronto, Canada
frank@spoclab.com
[6] AIAS Bologna Onlus, Bologna, Italy
{mmalavasi,ldesideri}@ausilioteca.org

Abstract. The paper introduces CloudCAST, a novel solution for making speech technology tools available to developers of speech-enabled applications, including intelligent tutoring systems (ITSs). The historical goal of fully integrating speech into ITSs is considered in the current context. Benefits of speech technology as they relate to ITSs are highlighted and a method for making these tools available to users with no speech technology expertise, through a remotely-located cloud-based platform is proposed. The challenges and opportunities are discussed with a view to reviving the interest of the developers of ITSs.

Keywords: Intelligent tutoring system · Speech technology
Speech recognition · Cloud-based speech tools · CloudCAST

The aim of this paper is twofold. Firstly, it puts forward the position that greater acceptance of intelligent tutoring systems can be achieved by increasing the incorporation of speech technology tools. Secondly, the paper seeks to introduce an ongoing effort to develop the CloudCAST platform. CloudCAST is a resource that will facilitate the inclusion of speech in ITSs by providing a range of customisable speech technology tools that can be deployed by speech technology experts, as well as users without a background in speech technology.

CloudCAST is an International Network (IN-2014-003) funded by the Leverhulme Trust.

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 421–424, 2018.
https://doi.org/10.1007/978-3-319-91464-0

It has long been the goal of researchers to develop an intelligent tutoring system (ITS) that provides automated, customised and adaptive feedback to learners. Recent advances in artificial intelligence (AI) has enabled the development of advanced ITSs that offer the personalisation of learning at scale.

Though there has been significant progress, and considerable success with ITSs in recent years, this has been somewhat limited by the exclusion of speech input from the user. An ITS with speech-enabled dialogue has several advantages, including: the provision of a more natural mode of communication for the learner, as well as the detection of learner understanding and engagement, which is gauged from the recognition of dysfluencies and learner affect.

Notwithstanding the obvious benefits to including a speech-enabled dialogue interface in ITSs there is still some resistance. This is mainly due to the historical performance of automatic speech recognition (ASR) systems [3], but also in part due to findings that suggest no added value to a speech interface [2, 4]. Many of the objections are based on the past failures of ASR systems, however, recent systems have been achieving remarkable performance, even in the most challenging domains, such as children's speech and achieving accuracy equal to humans on specific datasets. While it has been shown that errors in ASR transcription can ultimately lead to frustration with the ITS, analysis has shown that ASR errors do not negatively impact learning if the user persists with the tool, which is not a given.

This work makes the case for increasing the use of speech and language tools in ITSs and outline a solution for making this a less daunting prospect for ITS developers without a background in speech processing. It introduces the CloudCAST platform and shows how it can be beneficial to developers of ITSs that would like to exploit speech technology tools, but have no time or expertise to develop them.

The early visionaries were clear about the role of speech in the intelligent tutor. Two-way speech dialogues were seen as the ideal means of interaction between the tutor and learner [1, 5]. This goal was not achievable at the time, given the state of the art in speech technology, in particular, automatic speech recognition. Since then, remarkable improvements have been to ASR technology, which has advanced so significantly, that some commercial systems, such as Cortana, Siri, Google and Alexa, have become household names and directly impact our daily lives.

These developments in ASR technology have been generally ignored by the ITS community; with few exceptions, ITS systems do not include ASR technology. It is argued here that this should change, in part because research has found that having a completely spoken dialogue system provides significant gains to learning outcomes. It has been shown that: spoken communication between the tutor and learner does more to engage the learner and encourage constructivist learning; improvement in the student model, and the attendant improvement in learning outcomes, can be achieved using speech, and finally, a speech-based dialogue puts the student at ease, making the learning environment more social and comfortable.

The aim of the CloudCAST platform is to provide a suite of speech technology tools that can be employed in a wide variety of applications that require a speech interface. Developers of applications, including ITSs, would be able to use the provided tools, or develop bespoke speech recognition systems that can then be embedded in their applications. The platform is being developed in the cloud, with free access, where possible, to the tools and easy to use interfaces that will allow interested parties to use the tools with very little technical expertise required. The platform allows for multiple user types to access and make use of the tools provided. These groups include: Developers, who want to embed the technology into their own applications; end users, for whom applications are developed, e.g., children learning to read and speech technologists, who are improving or adding to the platform itself.

The platform can also be used by speech experts in order to collect speech data to build new recognition systems, if for instance a new technique is developed and the user wants to test it in their own application. Subject to ethical consent, interactions with the platform can be recorded. Thus the data that is collected can be used to retrain and improve the performance of the speech recognition tools over time.

Advantages of the Platform

There are other platforms that provide cloud-based recognition services, Speechmatics, Google's Web Speech API and SoundHound, for example. The challenge with these services include: the lack of customisation potential - the recognisers provided are the only ones that can be used, the limited output returned from the services and the lack of support for disordered and non-native speech.

By providing significant control over customisation and deployment, the proposed service will allow for personalised recognisers to be trained, or adapted, by the application developer and used by the learner. Functionality exists that will allow the user to record their data through the proposed platform, which will contribute to the effort to collect additional datasets, which will in turn assist in the effort to improve recogniser performance for end-users.

For such an ambitious platform, there are challenges - some progress has been made to date. These challenges are not insurmountable and can be overcome if there is significant buy-in from the technical community, who would be willing to contribute to the project in order to ensure its viability.

References

1. Carbonell, J.R.: AI in CAI: an artificial-intelligence approach to computer-assisted instruction. IEEE Trans. Man Mach. Syst. **11**(4), 190–202 (1970)
2. D'mello, S.K., Dowell, N., Graesser, A.: Does it really matter whether students' contributions are spoken versus typed in an intelligent tutoring system with natural language? J. Exp. Psychol. Appl. **17**(1), 1 (2011)
3. Litman, D.J., Forbes-Riley, K.: Speech recognition performance and learning in spoken dialogue tutoring. In: Interspeech, pp. 161–164 (2005)

4. Litman, D.J., Rosé, C.P., Forbes-Riley, K., VanLehn, K., Bhembe, D., Silliman, S.: Spoken versus typed human and computer dialogue tutoring. Int. J. Artif. Intell. Educ. **16**(2), 145–170 (2006)
5. Stevens, A.L., Collins, A.: The goal structure of a socratic tutor. Technical report 3518, Bolt Beranek and Newman INC., Cambridge, MA (1977)

Using E-learning System to Influence on User's Behavior Toward the Cybersecurity Strategy

Hasna Elkhannoubi(✉) and Mustapha Belaissaoui

Hassan I university, ENCG Information for decision laboratory, Settat, Morocco
h.elkhannoubi@uhp.ac.ma
http://encg-settat.ma

Abstract. Users behavior has consistently been reported as a key of effectiveness of the organization cybersecurity strategy. However, our interest is to understand the user's behavior and to influence on this behavior through an E-learning system. We opt to an Opinion Leader Agent O which exercises his influence on other users using an E-learning system as a communication vehicle. The findings suggest that social influences through an E-learning system play an important role in the cybersecurity strategy efficacy.

Keywords: Cybersecurity · User's behavior · E-learning
Social influence

1 Introduction

The global society is living in the electronic age where electronic transactions such as e-mail, e-banking, e-commerce and e-learning are becoming more and more prominent [4]. However, with this increasing proliferation of information and communication technologies, organizations should secure their cyber infrastructures through an effective cybersecurity strategy.

Given the background that, the cybersecurity strategy value is realized only when it is utilized by their intended users in a manner that contributes to its efficacy, the main purpose of the study is to strengthen the users integration in the cybersecurity strategy efficacy by proposing a new social influence model named Opinion Leader Influence (OLI) based on an E-learning system.

2 The E-learning System

E-learning is also called computer-based learning, on-line learning, distributed learning, or web-based training, has been defined differently in the literature. However, in this study we focus primarily on user's training development via network technologies, where the purpose is to increase user's knowledge and skills by influencing on its behavior using the referents influence: the influence of a minority of members in an organization possess qualities that make them

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 425–427, 2018.
https://doi.org/10.1007/978-3-319-91464-0

exceptionally persuasive in spreading ideas to others in a specific context. Hence, the E-learning system in our case can be defined as the development of the information security background of user's through awareness by using information technologies to bring out instructions and information to the organization's employees. By farther, the instructors, lecturer or content creator is an influential agent named Opinion Leader Agent O.

3 The Social Influence Theories

In one hand, the present study is influenced by the Social Cognitive Theory (SCT) which is an empirically validated theory of individual behavior based on Bandura's [1] work. In the other hand, our contribution is relay on the social power model developed by [2]. The social power of O/P in some system S is defined as the maximum potential ability of the element O to influence on the element P in S [2]. In our context, the agent O has an influence on the user P and applies a social power through an E-learning system.

4 The Opinion Leader Influence Through an E-learning System

In this study we hope to define the influence of an agent O named *Opinion Leader Agent* on a simple user P, where the communication vehicle between the agent O and P is an E-learning system developed by the agent O in a way to spread some ideas, instructions and policies. Our model of social influence is represented in Fig. 1:

The opinion leader agent O produces a social influence on the simple users P_1, P_2, P_3 and P_4, therefore, O is able to induce a strong force on P_i to carry out an activity related to the security of the organization's information system. This influence induced by O don't includes P_i's own forces because we assume that P_i is totally open to the O's influence. At this point, we assumed that O is capable to exert this influence on P_i because of some characteristics which he possesses.

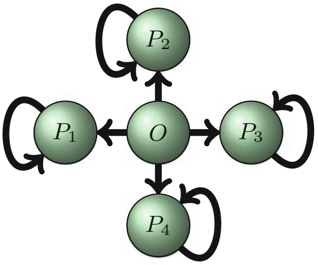


Fig. 1. The interaction graph (1)

The opinion change in structures of influential communication as presented by Friedkin [3] describes the process of opinion change that occurs among the members of a population about particular matter as formulated in the equation (1), where $m_{i(t+1)}$ is the opinion of member i at time $t+1$, N is the number of members of the population, w_{ij} is the weight member i accords to the opinion of member j, in other words is the effect of member j's opinion on member i's opinion:

$$\forall i \in \mathbb{N}^{*+}, m_{i(t+1)} = \sum_{j=1}^{N} w_{ij} m_{j(t)} \tag{1}$$

To validate our model, we work with the special case where the opinion leader agent O accords some influence to P_i ($w_{op_i} > 0$); however, P_i accords no influence to O ($w_{p_i o} = 0$). In this case our model will be showing in the equation (2)

$$P_{i(t+1)} = w_{pp} P_{i(t)} + w_{op_i} O_t \tag{2}$$

P_i isn't a stubborn users ,so, we ignore the weight accorded to P_i's opinion and we attache no weight to its own opinion ($w_{pp} = 0$) because O exerts a power on P_i and possesses all characteristics which make P_i open to its influence. In such case the equation (2) simplifies to:

$$P_{i(t+1)} = w_{op_i} O_t \tag{3}$$

To summarize, we assume that after some units of time the opinion of P_i will be totally influenced by the opinion of O and of course the behavior of P_i will be influenced by the behavior of O, so, we can benefit from a successful cybersecurity strategy if we create a social influence environment by the integration of the opinion leaders agents who monitors an E-learning system into the organization.

5 Conclusion

The social influence model serves to enhance the users' participation to the organization information security through their compliance to the cybersecurity strategy. Theoretically, the finding in this study suggests that social influence plays an important role in the efficacy of the cybersecurity strategy through the implementation of an E-learning system monitoring by an Opinion Leader Agent.

References

1. Bandura, A.: Social Foundations of Thought and Action: A Social Cognitive Theory. Prentice-Hall, Inc. (1986)
2. Bandura, A.: Social cognitive theory of mass communication. Media Psychol. **3**(3), 265–299 (2001)
3. Friedkin, N.E.: A formal theory of social power. J. Math. Sociol. **12**(2), 103–126 (1986)
4. Kritzinger, E., Von Solms, S.: E-learning: Incorporating information security governance. Issues in Inf. Sci. Inf. Technol. **3** (2006)

Evolution of Methods of Evaluation in AI in Education

Reva Freedman

Northern Illinois University, Dekalb, IL, USA
rfreedman@niu.edu

Abstract. A study of methods of evaluation in the field of AI in Education shows great changes in a 15-year period. The percent of papers in two major conferences that include some type of numerical evaluation, whether statistical or not, increased from 6% in ITS 1996 to 94% in ITS 2010. This differs from the pattern in the AAAI conference, which started with a higher baseline but increased more gradually.

Keywords: History of artificial intelligence · History of science Intelligent tutoring systems · Artificial intelligence in education

Given the theme of ITS 2018, "A 30 Year Legacy of ITS Conferences," we have chosen to study changes over the lifetime of the field of AI in Education. In this paper we show that the field has undergone a major change with respect to methods of evaluation in the 15-year period from 1996 to 2010. To demonstrate this, we analyze the proceedings of two major conferences in the field over that time period, including ITS 1996 [1], ITS 1998 [2], ITS 2000 [3], AIED 2001 [4], AIED 2003 [5], ITS 2004 [6], AIED 2007 [7] and ITS 2010 [8]. We show that the percent of papers published in major annual conferences that contain a statistical test has increased significantly in the last two decades.

We counted the number of papers in the main session of each of the selected conferences. We did not count invited talks, posters, papers in the student session or workshop papers. We divided the selected papers into three categories: papers that contained at least one statistical test, papers that contained a numerical evaluation but no statistical tests, and papers that contained neither. Although a few of the latter contained a mathematical derivation, including mathematical logic, most did not; rather, they were descriptive papers. We counted as containing a numerical evaluation any paper that had collected data on two or more categories where it would be possible to do a valid statistical test.

The results are shown in Table 1. The Stat column contains the number of papers containing a statistical test, while the Eval column contains the number of papers containing some kind of numerical evaluation, whether statistical or not. The column labeled None equals 100% minus the Eval column.

The results are shown graphically in Fig. 1. Each column of the graph shows a percentage breakdown of the papers published in that year. For clarity, the Eval section of each bar shows the percent of papers containing only a non-statistical evaluation.

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 428–430, 2018.
https://doi.org/10.1007/978-3-319-91464-0

Table 1. Results from Selected AI and Education Conferences

AI and Ed Conf.	Total	Stat	%	Eval	%	None	%
ITS 1996	69	3	4	4	6	65	94
ITS 1998	59	5	8	10	17	49	83
ITS 2000	61	6	10	15	25	46	75
AIED 2001	45	8	18	14	31	31	69
AIED 2003	40	15	38	18	45	22	55
ITS 2004	72	25	35	37	51	35	49
AIED 2007	60	37	62	46	77	14	23
ITS 2010	61	50	82	57	93	4	7

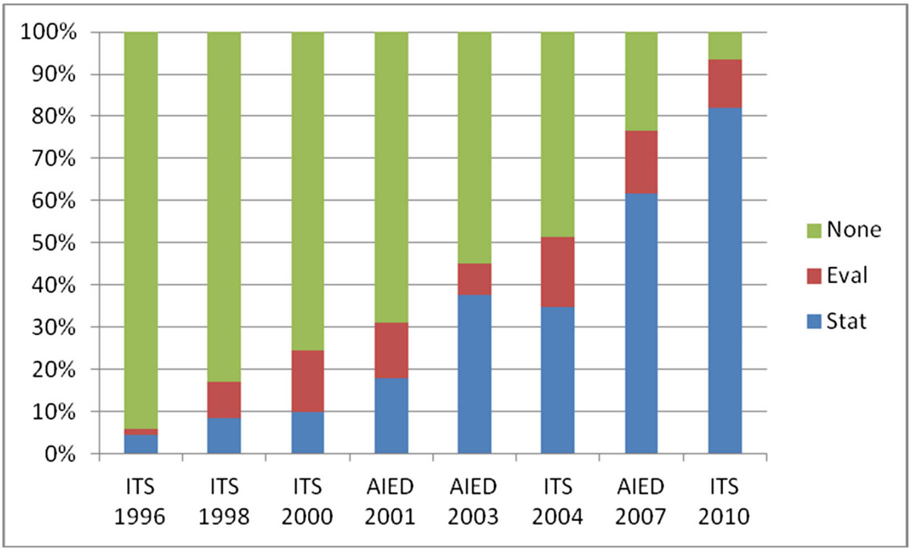


Fig. 1. Results from Selected AI and Education Conferences

These data show the increasing importance of a numerical evaluation, preferably a statistical one, in the field of AI and Education. In an earlier paper looking forward to the year 2010, Cumming and McDougall state that “in the early days AIED was to some extent computer scientists at play” [9]. This quote is borne out by the data we analyzed. As further evidence for a shift in emphasis, in the early proceedings, there was even an occasional paper in the “Evaluation” section that contained no statistics.

For comparison purposes we did the same evaluation on three annual conferences sponsored by the Association for the Advancement of Artificial Intelligence (AAAI) in the same timeframe, AAAI 1996 [10], AAAI 2002 [11], and AAAI 2010 [12]. Papers were analyzed using the same criteria as the AI and Education papers. Since those conferences were larger, we attempted to select a random subset of papers to analyze. To this end we used the first paper listed under each subtopic in the table of contents. The result of this study are shown in Table 2.

Table 2. Results from Selected Artificial Intelligence Conferences

AI Conference	Total	Stat	%	Eval	%	None	%
AAAI 1996	49	6	12	32	65	17	35
AAAI 2002	16	0	0	11	69	5	31
AAAI 2010	10	0	0	9	90	1	10

The AAAI data also show that the percent of papers containing no numerical evaluation have dropped, but the baseline is smaller, so the decline is less severe. In addition, almost all of the AAAI papers that contained no numerical evaluation contained proofs, mostly logic proofs but a few numerical ones. Conversely, while a few of the AI and Education papers contained a derivation using mathematical logic, almost none contained proofs.

In future work, we would like to evaluate three hypotheses for these changes in the field of AI and Education. Have individual projects moved from a planning stage to an evaluation stage, i.e., what the early proceedings show is the startup phase of the field? Or have publication criteria changed, so that a formal evaluation is required for publication as a full paper? Or has researcher interest moved from non-numerical methods to numerical ones?

References

1. Frasson, C., Gauthier, G., Lesgold, A. (eds.): ITS 1996. LNCS, vol. 1086. Springer, Heidelberg (1996). https://doi.org/10.1007/3-540-61327-7
2. Goettl, B.P., Halff, H.M., Redfield, C.L., Shute, V.J. (eds.): ITS 1998. LNCS, vol. 1452. Springer, Heidelberg (1998). https://doi.org/10.1007/3-540-68716-5
3. Gauthier, G., Frasson, C., VanLehn, K. (eds.): ITS 2000. LNCS, vol. 1839. Springer, Heidelberg (2000). https://doi.org/10.1007/3-540-45108-0
4. Moore, J., Redfield, C., Johnson, W. (eds.): Proceedings of the Tenth International Conference on Artificial Intelligence in Education (AIED 2001). IOS Press (2001)
5. Hoppe, U., Verdejo, F., Kay, J. (eds.): Proceedings of the 11th International Conference on Artificial Intelligence in Education (AIED 2003). IOS Press (2003)
6. Lester, J.C., Vicari, R.M., Paraguaçu, F. (eds.): ITS 2004. LNCS, vol. 3220. Springer, Heidelberg (2004). https://doi.org/10.1007/b100137
7. Luckin, R., Koedinger, K., Greer, G. (eds.): Proceedings of the 13th International Conference on Artificial Intelligence in Education (AIED 2007). IOS Press (2007)
8. Aleven, V., Kay, J., Mostow, J. (eds.): ITS 2010. LNCS, vol. 6094. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13388-6
9. Cumming, G., McDougall, A.: Mainstreaming AIED into education. Int. J. Artif. Intell. Educ. **11**, 197–207 (2000)
10. Proceedings of the Thirteenth National Conference on Artificial Intelligence. AAAI Press (1996)
11. Proceedings of the Eighteenth National Conference on Artificial Intelligence. AAAI Press (2002)
12. Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence. AAAI Press (2010)

Typing-Differences in Simultaneous Typed Chat

Michael Glass[1](✉), Yesukhei Jagvaral[1], Chinedu Emeka[2], and Jung Hee Kim[3]

[1] Valparaiso University, Valparaiso, IN, USA
michael.glass@valpo.edu
[2] University of Illinois, Urbana, IL, USA
[3] North Carolina A&T State University, Greensboro, NC, USA

Abstract. Full-duplex conversation where everybody can talk and hear at the same time is made possible by typed-chat computer-mediated communication. This experiment examines typing logs from students engaging in overlapping dialogue chat in small-group problem-solving sessions. When students are typing in the presence of overlapping dialogue there are measurable differences in their typing behavior. A difference measured here is text-deletion behavior. Deletions increase in the simultaneous typing regime. The reasons for this difference remain to be explored.

Keywords: Typed-chat · Full-duplex dialogue

1 Background

1.1 Introduction

The COMPS project deploys and studies small-group collaborative problem-solving exercises in college computer science and mathematics classes [3]. A striking feature of the chat environment is it permits everybody to type and see and respond to each other's dialogue all at the same time. Full duplex typed computer chat differs from ordinary computer chat [4]. Effectively there is no such thing as interruption. A second person starting to type contributes to the conversation immediately, but in no way affects the first person's ability to type.

How students utilize this non-natural mode of communication when collaborating in a problem-solving dialogue is still relatively unexplored. The hypothesis considered in this paper is that since the communication medium does not impede simultaneous chatting in the same way that person-to-person talking does, chat behaviors won't differ compared to when a single person has the floor.

This paper shows one measurable difference in editing behavior. Students delete text more often when there are other students typing.

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 431–433, 2018.
https://doi.org/10.1007/978-3-319-91464-0

1.2 COMPS Exercises

COMPS small-group problem-solving exercises [3] are designed to address student conceptual knowledge through group cognition. The problems for discussion typically have many parts, often with multiple-choice answers. The exercise protocol discourages social loafing by requiring students to come to agreement at various points in the conversation. There is an answer window where the students construct an answer explanation for the TA, who must check it. The TA then engages with the students via the typed-chat conversation, and assists if they are off track [3].

1.3 Simultaneous Chat

Allowing everybody to chat simultaneously could potentiate student engagement, as it isn't necessary to wait for one's fellow students to relinquish the floor before contributing one's own thoughts into the discussion. Allowing everybody to chat simultaneously also should discourage social loafing, one student cannot dominate the conversation by aggressively interrupting others. However the possibility exists that absent enforced turn-taking, full-duplex communication enables students to ignore each other and forego transactive conversation.

Earlier work from the COMPS project has shown that in the simultaneous typing regime students still engage in transactive turn-taking conversational behaviors where they respond to each other [1]. Interactions commonly take several forms [2], viz:

1. Student B responds to something that A just said, while A continues uninterrupted.
2. Students B and C both respond to student A's utterance.
3. Students A and B utter unrelated dialogue turns, each continuing earlier discourse threads by possibly other people.

What these behaviors have in common is a student does not need to respond to the other person's keystrokes in real time. An utterance usually responds to keystrokes that happened before the utterance commenced. The novel medium of communication therefore does not, in this aspect, produce novel discourse behaviors different from the Initiate/Respond/Follow-up structure discovered by Conversation Analysis [5].

2 Experiment and Discussion

The data for this study were 56 small group conversations in a Java class of approximately one hour each. Almost all conversation groups had 3 students, with one TA or professor attending to the conversation part-time. Keystroke log records were separated into those that occur when one person is typing (the "alone" condition) and when several people were typing (the "simultaneous" condition). 2.0 s time separation from all other participants was needed to characterize a keystroke as "alone." Table 1 summarizes the results of tabulating deletion and non-deletion keystrokes in the alone and simultaneous conditions. Considering overall averages among all participants in all conversations, deletions increased from 8.9% to 13.9% of keystrokes when other people were typing. A two-tailed pairwise Student's t-test showed the difference was significant, with $p < 0.001$. The data were also analyzed as 163 separate pairwise comparisons, each comparison representing the behavior of one person in one conversation who had contributed at least 80 keystrokes in both the alone and simultaneous conditions. Paired t-test also showed deletions were significant with $p < 0.001$.

Table 1. Deletions as a fraction of all keystrokes, typing alone and simultaneously.

N = 56 dialogs	Alone	Simultaneous
Keystrokes total	246274	47890
Mean keys/dialogue	4398	855
Deletion fraction	0.089	0.139
Std. Dev (N = 56)	0.034	0.077

We have yet to explore whether one student's increased deletions licenses other student to start simultaneous dialogue, or whether the presence of other students on the conversational floor permits one to spend more time editing. Earlier work showing that pauses are transition-relevance points [5] permitting turn-taking suggests the former is likely [2]. In addition, we have found changes in typing speeds which vary by individuals, so it is quite possible that deletion behaviors vary by individuals also. Correlating full-duplex dialogue behaviors with transactive dialogue moves also remains to be done.

Acknowledgment. Partial support for this work was provided by the National Science Foundation's Improving Undergraduate STEM Education (IUSE) program under Award No. 1504918.

References

1. Glass, M., Kim, J.H., Bryant, K., Desjarlais, M.: Come let us chat together: simultaneous typed-chat in computer-supported collaborative dialogue. J. Comput. Sci. Coll. **31**(2), 96–105 (2015)
2. Glass, M., Nelson, A., Emeka, C., Kim, J.H.: Not interfering: simultaneous typed chat in COMPS computer-mediated dialogues. In: 28th Modern AI and Cognitive Science Conference, pp. 107–113, Fort Wayne, IN (2017)
3. Kim, J.H., Kim, T., Glass, M.: Early experience with computer supported collaborative exercises for a 2nd semester java class. J. Comput. Sci. Coll. **32**(2), 68–86. (2016)
4. Paolillo, J.C., Zelenkauskaite, A.: Real-time chat. In: Herring, S., et al. (eds.) Pragmatics of Computer-Mediated Communication, pp. 109–133. Mouton de Gruyter (2013)
5. Sacks, H., Schegloff, E., Jefferson, G.: A simplest systematics for the organization of turn-taking for conversation. Language **50**(4), 696–735 (1974)

Examining How Students' Typical Studying Emotions Relate to Those Experienced While Studying with an ITS

Jason M. Harley[1(✉)], François Bouchet[2], and Roger Azevedo[3]

[1] University of Alberta, Educational Psychology, Edmonton, AB, Canada
jharley1@ualberta.ca

[2] Laboratoire d'Informatique de Paris 6, LIP6, Sorbonne Université, CNRS,
75005 Paris, France
francois.bouchet@lip6.fr

[3] North Carolina State University, Psychology, Raleigh, NC, USA
razeved@ncsu.edu

Abstract. We help advance the research on emotions with a preliminary investigation of differences between 116 students' typical studying emotions and those they experienced while studying with an ITS. Results revealed that students reported significantly lower levels of negative emotions while studying with an ITS compared to their typical emotional dispositions toward studying.

Keywords: Emotions · Affect · Intelligent tutoring systems
Pedagogical agents

1 Introduction

Achievement emotions are critical because of the impact they have on our success and failure in important and influential domains such as learning and success in school [1]. Emotions can support achievement by fostering motivation, focusing attention and limited cognitive resources on achievement-related activities and promoting adaptive information processing and self-regulation strategies [1]. While research has focused on the emotions learners tend to experience while interacting with these systems, little is known about how students general academic emotional tendencies compare with those experienced during these, often novel, interactions [2]. Understanding how students typically feel while studying is valuable because of its potential to inform user models and design more adaptive ITSs [3]. Moreover, comparisons provide an affective benchmark to help researchers appreciate affective benefits or shortcomings that systems have when compared to students' academic status quo.

In this study, we investigated the effect of administering the achievement emotions questionnaire (AEQ [1]) prior to learners' interaction with MetaTutor and halfway through their interaction with it on the negative emotions they reported experiencing. We were particularly interested in learners' negative emotions because of the deleterious impact they can have on learners' experience with the system, self-regulated learning skill use, and achievement. Our hypothesis was that learners would report

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 434–437, 2018.
https://doi.org/10.1007/978-3-319-91464-0

lower intensity levels of these emotions while studying with MetaTutor on account of lower appraisals of instrumental task value [4, 5]. In other words, because MetaTutor is a low stakes studying environment, like many ITSs, students can focus on content and process practice and mastery without concern for grades [3].

2 Methods

2.1 Participants and Experimental Conditions

One hundred and sixteen undergraduate students ($N = 116$, 17–31 years old, $M = 20.9$ years, $SD = 2.4$; 64.6% female; 62.9% Caucasian) from two North American Universities, studying different majors and with various levels of prior knowledge participated in this study. Each participant received $50 upon completion of the study.

2.2 The ITS, Experimental Procedure, Measures and Data Sources

System Overview. MetaTutor [5, 6] is an ITS where four pedagogical agents (PAs) help students learn by prompting them to engage in SRL processes. A table of contents links to 38 pages (with text and images) on the human circulatory system.

Experimental Procedure. The experiment involved two different sessions separated by one hour to three days. During the first one (30 to 40 min. long), participants filled and signed a consent form and completed the AEQ trait questionnaire (see below), a demographics survey, and a pre-test on the circulatory system. During the second session (90 min. long), participants used MetaTutor to learn about the circulatory system. Participants had exactly 60 min to interact with the content during which they could initiate SRL processes or do so after a PA's prompt. After MetaTutor offered students a 5 min break (halfway through), it asked them to fill out the 'during studying state' emotion subscale of the AEQ. At the end of the session, participants were given a post-test. All participants completed their sessions individually.

Measures. The during studying trait emotions subscale (AEQ [1]) was used to measure the emotions learners' typically experience while studying. This AEQ subscale consists of 45 items and measures anger (5 items; $\alpha = .81$), anxiety (6 items; $\alpha = .78/.81$), shame, (7 items; $\alpha = .85/.89$), hopelessness, (5 items; $\alpha = .86/.91$), boredom, (9 items; $\alpha = .89/.94$). The same questionnaire was administered following the optional pause a second time, with changes in wording (based on [1, 2]) to assess the emotions learners experienced while they interacted with MetaTutor. Cronbach's Alpha indicated that internal reliability was acceptable for each subscale (admin 1/admin 2) for both administrations of the AEQ.

3 Results

Five paired sample t-tests were run to examine whether significant differences existed between learners' typical emotions experienced during studying (AEQ 1) and the emotions they reported while studying with MetaTutor (AEQ 2). Outlier screening was performed and outlying scores were replaced with the next most extreme score. AEQ 1 and 2 differed significantly for all negative emotions: anger, anxiety, shame, hopelessness, and boredom. Specifically, emotions were higher during typically studying session than learners' interaction with MetaTutor (see Table 1).

Table 1 Learners' emotions during typical vs. MetaTutor studying sessions

AEQ Variable				AEQ1		AEQ2	
	t	*df*	*p < .05*	*M*	*SD*	*M*	*SD*
Anger	5.13	112	p < .05	2.18	.81	1.99	.81
Anxiety	10.40	112	p < .01	3.02	.87	2.15	.83
Shame	8.70	112	p < .01	2.56	.96	1.84	.82
Hopelessness	2.70	112	p < .01	1.87	.85	1.61	.70
Boredom	3.04	112	p < .01	2.55	.81	2.27	1.00

4 Discussion

Results supported our hypothesis that achievement emotions reported during learners' interactions with MetaTutor would be lower in intensity than those reported beforehand that reflected how learners typically felt while studying. Experiencing lower levels of negative activating and de-activating emotions tends to be beneficial to students' academic achievement. Future research should examine learners' appraisals of value and their relationships to achievement emotions in typical academic achievement situations (e.g., studying) versus interactions with ITSs.

Acknowledgements. Research supported by funding from NSF (DRL 1008282, DRL1431552, DRL 1660878), SSHRC, and CRC program awarded to third author.

References

1. Pekrun, R., Goetz, T., Titz, W., Perry, R.P.: Academic emotions in students' self-regulated learning and achievement: a program of quantitative and qualitative research. Educ. Psychol. **37**, 91–106 (2002)
2. Harley, J.M., Carter, C.K., Papaionnou, N., Bouchet, F., Azevedo, R., Landis, R.L., Karabachian, L.: Examining the predictive relationship between personality and emotion traits and students' agent-directed emotions: towards emotionally-adaptive agent-based learning environments. User Model. User-Adap. Inter. **26**, 177–219 (2016)
3. Harley, J.M., Lajoie, S.P., Frasson, C., Hall, N.C.: Developing emotion-aware, advanced learning technologies: a taxonomy of approaches and features. Int. J. Artif. Intell. Educ. **27**(2), 268–297 (2017)
4. Harley, J.M., Bouchet, F., Hussain, S., Azevedo, R., Calvo, R.: A multi-componential analysis of emotions during complex learning with an intelligent multi-agent system. Comput. Hum. Behav. **48**, 615–625 (2015)

5. Azevedo, R., Martin, Seth A., Taub, M., Mudrick, Nicholas V., Millar, Garrett C., Grafsgaard, Joseph F.: Are pedagogical agents' external regulation effective in fostering learning with intelligent tutoring systems? In: Micarelli, A., Stamper, J., Panourgia, K. (eds.) ITS 2016. LNCS, vol. 9684, pp. 197–207. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39583-8_19
6. Harley, J.M., Taub, M., Azevedo, R., Bouchet, F.: "Let's set up some subgoals": understanding human-pedagogical agent collaborations and their implications for learning and prompt and feedback compliance. IEEE Trans. Learn. Technol. **11**(1), 54–66 (2018)

Examining How Typical Gaming Behavior Influences Emotions and Achievement During Gameplay

Jason M. Harley[1](✉), Mohamed S. Benlamine[2], Maher Chaouachi[2], Claude Frasson[2], Yang Liu[1], and Aude Dufresne[2]

[1] University of Alberta, Educational Psychology,
Edmonton, AB, Canada
jharley1@ualberta.ca
[2] Université de Montréal, Computer Science and Operations Research,
Montréal, QC, Canada

Abstract. This study examined the effect of the quantity of weekly time 20 undergraduate students' spent gaming on their performance, physiological activation, and self-reported emotions while playing a game. Results revealed that the average number of hours an individual spent playing games a week influenced their physiological activation. Implications for educational games are discussed.

Keywords: Emotions · Affect · Games · Serious games · Physiological data

1 Introduction

In serious games, one of the most important individual differences between learners is their prior gaming experience. Educational games can take a wide variety of forms, ranging from immersive, 3D virtual worlds to 2D puzzle games [1]. Prior gaming experience influences learners' experience with educational games because it can provide them with procedural knowledge of game mechanics and influence their learning trajectory, cognitive load, and mediate their emotional experiences with the game [2]. Moreover, significant weekly investments in gaming are illustrative of high intrinsic value, which can also influence one's emotions during learning with serious games [3]. Understanding individual differences is critical to inform user models and design more adaptive, emotionally-aware systems [1]. In this study, we investigated the effect that prior gaming experience had on users' performance, emotional activation, and self-reported emotions while they played a videogame. We hypothesized that gamers who spent the greatest number of hours gaming a week would experience the highest levels of emotional intensity because of investment in game achievement (i.e., high appraisals of intrinsic value; [3]).

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 438–441, 2018.
https://doi.org/10.1007/978-3-319-91464-0

2 Methods

2.1 Participants and Prior Gaming Experience

Twenty undergraduate students (90% male; 50% Caucasian) and self-reported gamers from the computer science department of a North American University participated in this study. Participants had a mean age of 23.55. Participants were classified as follows: a casual gamer played 5-hours or less a week (*N* = 7); a heavy gamer played more than 15 (*N* = 7); middle were classified as moderate gamers (*N* = 6).

2.2 Experimental Procedure and Game

The study took approximately one hour and involved participants interacting with Assassin's Creed: Unity, a game developed by Ubisoft. During the session participants filled out a consent form, put on a Q-Sensor 2.0 bracelet (EDA), were introduced to the game console (Xbox One) and controls, and played through a tutorial (approx. 6 min). Users then watched a short in-game cut scene (movie) introducing the protagonist reminiscing about the pocket watch his father gave him just before being assassinated. The movie ends with the theft of the beloved pocket watch by a villain (Hugo). The gameplay the study focused on was capturing Hugo and retrieving the watch. Once users caught Hugo the game was paused and they filled out an emotion questionnaire. Users were compensated $20 for participating at the end.

2.3 Measures, Materials, and Scoring

Achievement Measure. Achievement was measured as the time it took participants to catch Hugo. It took players 1 min and 12 s on average to capture him (*SD* = 30 s); the fastest player took 39 s and the slowest took nearly 3-min.

Self-report Measure. Users completed an emotion questionnaire (see Table 1) immediately following Hugo's capture. Items were based on [4] and assessed using a 5-point Likert scale ranging from "Strongly Disagree" to "Strongly Agree."

Table 1. Descriptive statistics

Variable	Gaming level					
	Casual		*Moderate*		Heavy	
	M	SD	*M*	*SD*	*M*	*SD*
Achievement	76.29	29.35	55.83	3.98	65.14	1.50
Physiological arousal	0.40	0.26	0.11	0.01	0.57	0.28
Enjoyment	4.71	0.49	4.33	0.52	3.57	0.98
Frustration	2.00	1.15	1.50	0.55	2.00	1.15
Anxiety	2.71	0.49	2.50	1.38	2.14	1.07
Boredom	1.29	0.49	2.00	1.26	2.00	1.15

Q-Sensor 2.0 EDA Bracelet and Physiological Data. Q-Sensor 2.0 was used to measure players' electrodermal activity (EDA); a signal commonly used to measure physiological arousal. Measurements are understood in relative terms due to individual differences in baseline EDA levels. Q-Sensor data corresponded to the analyses of the 10 s prior to capturing Hugo. EDA values were normalized on a 1–10 scale based on a user-dependent model that took participants' individual EDA ranges into consideration. Normalized EDA values were between 0 and 1 and EDA scores were interpreted based on proximity to these extremes. Procedures were based on [4].

3 Results

A one-way ANOVA was run to examine the effect of users' prior gaming on their achievement (time taken to catch Hugo), but failed to reveal a significant effect. A one-way ANCOVA was run to examine the effect of users' prior gaming experience on their physiological activation while controlling for achievement. A significant main effect of users' prior gaming experience was observed, $F(2, 16) = 6.78$, $p < .01$, $n^2p = .46$. Pairwise Bonferonni comparison tests revealed a significant difference between heavy gamers' ($M = .57$; $SD = .28$) and moderate gamers ($M = .11$; $SD = .01$) physiological activation (see Table 1). Data screening revealed that the self-reported emotions were best examined descriptively in this study (see Table 1).

4 Discussion

Results confirm that prior gaming experience influences users' physiological activation and that heavy gamers—who spent a significant portion of their weeks playing games—had the highest levels of physiological arousal. Heavy gamers did not, however, report the highest levels of discrete emotions; casual gamers did. This effect may be due to heavy gamers not wanting to acknowledge how they felt in an achievement task that was important to them, whereas casual gamers may have had fewer qualms with acknowledging their emotions—for better and worse—with a game. Moderate players may have had the lowest levels of arousal because they were comfortably competent, but not overly invested in their performance. Further research should examine users' motivational goal orientations to test this interpretation. These results highlight the influence of prior gameplay on users' emotions and achievement—individual differences that emotionally-adaptive systems can leverage.

Acknowledgements. Research supported by a postdoctoral fellowship from the Fonds Québécois de recherche–Société et culture (FQRSC) awarded to first author.

References

1. Harley, J.M., Lajoie, S.P., Frasson, C., Hall, N.C.: Developing emotion-aware, advanced learning technologies: a taxonomy of approaches and features. Int. J. Artif. Intell. Educ. **27**(2), 268–297 (2017)
2. Mayer, R.E. (Ed.).: The Cambridge Handbook of Multimedia Learning, 2nd edn. Cambridge University Press, New York (2015)
3. Pekrun, R., Perry, R.P.: Control-value theory of achievement emotions. In: International Handbook of Emotions in Education, pp. 120–141. Routledge, NY (2014)
4. Harley, J.M., Bouchet, F., Hussain, S., Azevedo, R., Calvo, R.: A multi-componential analysis of emotions during complex learning with an intelligent multi-agent system. Comput. Hum. Behav. **48**, 615–625 (2015)

The Design of a Learning Analytics Pedagogical Dashboard to Enhance Instructors' Facilitation in an Online Asynchronous Problem-Based Learning Environment

Lingyun Huang[1](✉), Stephen Bodnar[1], Juan Zheng[1],
Maedeh Assadat Kazemitabar[1], Yuxin Chen[2], Gurpreet Birk[2],
Cindy E. Hmelo-Silver[2], and Susanne P. Lajoie[1](✉)

[1] McGill University, Montreal H3A 1Y2, Canada
{lingyun.huang, juan.zheng, maedeh.kazemi}@mail.mcgill.ca, {stephen.bodnar2, susanne.lajoie}@mcgill.ca
[2] Indiana University Bloomington, Bloomington, IN 47405-1006, USA
yc58@iu.edu, gurpreet.s.birk@gmail.com, chmelosi@indiana.edu

Abstract. Problem-based learning (PBL) refers to small group collaborative learning situations where students solve complex problems with the assistance of teachers who serve as facilitators. Scaling PBL using technology requires specific tools since online asynchronous PBL can increase the number of small groups that engage in the curriculum but poses challenges to PBL teachers who must attend to multiple groups. To address the problem, we have been researching how technology can be used to develop specific tools to extend expert teachers' instructional capacities. Building on previous work, we present the most recent design of a pedagogical dashboard used in an online asynchronous PBL environment. We illustrate how the new pedagogical dashboard visualizations can support PBL instructors observing individual student learning activities, diagnosing group dynamics and intervening when necessary.

Keywords: Pedagogical dashboard · Learning analytics · Visualizations
Online asynchronous PBL

1 Problem Statement and Intended Goals

Problem-based learning (PBL) is an instructional design in which a group of students co-direct and co-regulate their learning efforts and processes to address ill-structured problems (Hmelo-Silver, 2004). Shifting focus to online contexts, PBL researchers find asynchronous online PBL expands student participation and allows students to communicate beyond the boundary of different geographical limitation and to provide in-depth, more thoughtful discussion (Lajoie et al., 2014). However, it challenges PBL instructors, increasing their workload and requiring a higher pedagogical capacity to monitor and diagnose student activities and interaction. The primary concern of the

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 442–445, 2018.
https://doi.org/10.1007/978-3-319-91464-0

challenges is that PBL instructors have difficulty obtaining information concerning students' learning that is critical for making pedagogical decisions and provide appropriate facilitation. To solve the problem, we propose a learning analytics pedagogical dashboard (LAPD) that intends to be an instructional tool analyzing student-generated data to inform students' learning actions and group dynamics to instructors in online PBL contexts. The following sections will present the techniques we applied in the LAPD together with our iterative design process.

2 The Learning Analytics Pedagogical Dashboard

The LAPD incorporates learning analytics and visualization techniques to effectively present information related to student learning. Learning analytics involves tracking and analyzing a collection of student-generated data and metadata through sophisticated analytical techniques with the purpose to identify students' actions and patterns (Ferguson, 2012). The data can be visibly displayed through multiple visualization tools. In our case, the LAPD is implemented in online asynchronous PBL which an instructor facilitates multiple groups of medical students (Hmelo-Silver et al., 2016; Lajoie et al., 2014). Two types of student data are critical for the facilitation. The data capturing individual actions (e.g. the number of one's posts and comments, the frequency of words appearing in group discussion). In terms of understanding the group dynamics, the dashboard centers on the data related to the conversation like the progression, the direction, the number of conversation turns, and the frequency of the chats between two students. Similarly, we apply different visualization tools capturing student individual actions (e.g. task progress bar) and interaction (e.g. social network analysis chart).

We have made efforts to design and test the LAPD in the previous work. Despite the positive perceptions regarding the dashboard ability to support instructors' facilitation, the test also made clear that a number of improvements were needed (refer to Hogaboam et al., 2016, Kazemitabar et al., 2016). The second version (Fig. 1) incorporates Conversation Explorer, Social Network (SNA) View, Task Progress View and Activity View. Conversation Explorer, located at the top of the dashboard, visualizes students' participation and interaction over time. The Conversation Exploration illustrates group member interactions to gain insight into how a conversation developed. Meanwhile, data related to conversation contents are processed to generate a list of frequently used words in conversations. The SNA view located underneath the Conversation Explorer, contains several color-coded nodes and lines with arrow heads. Node size represents the amount of textural output students produced in the discussion, for example, the number of posts. The thickness of line suggests the extent to which students converse with one another. The arrow head indicates the information flow. The SNA view can illustrate group dynamics, for example, arrows flowing from larger nodes to smaller nodes can indicate when a given student may be dominating the discussion, the Task Progress View, relying on task completion, is illustrated in the form of grids. The visualization. Once a task is submitted, the corresponding cell in the grid will be highlighted in green. The Activities View on the bottom right. The x-axis in the chart is time and the y-axis is a frequency count of either chat posts or the word counts from these posts. Each point on the chart is obtained by averaging over a 12-hour period. The word count could indicate one's attention to learning process. We replace the pie chart with the line graph because it is more intuitive to indicate the developing process over time. Chat turns can reflect student participation. Student with lower numbers of chat turns may be more likely to spend less time on learning or almost drop out. This visualization can flag these students to prompt instructors to investigate further.

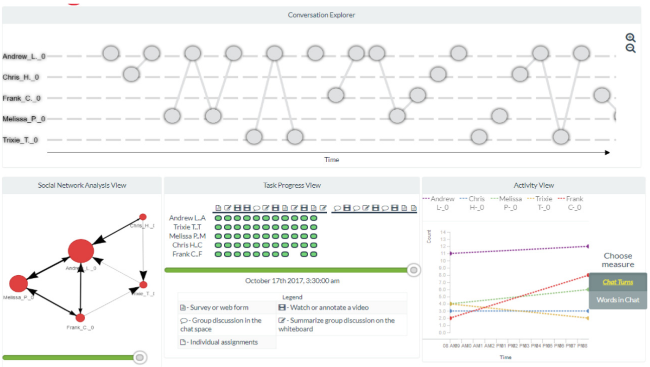


Fig. 1. The second version of the learning analytics pedagogical dashboard

3 Conclusion

The paper presents the design of the LAPD used for online asynchronous PBL. The iterative design process leads us to adjustments to strengthen its values of enhancing PBL instructors' facilitation. To test the robustness of the new design, we will analyze think-aloud data collected recently to understand instructors' perceptions. In future, we plan to test the dashboard with real students in an authentic world setting.

References

Ferguson, R.: Learning analytics: drivers, developments and challenges. Int. J. Technol. Enhanced Learn. **4**(5–6), 304–317 (2012)

Hmelo-Silver, C.E.: Problem-based learning: what and how do students learn? Educ. psychol. Rev. **16**(3), 235–266 (2004)

Hmelo-Silver, C.E., Jung, J., Lajoie, S., Yu, Y., Lu, J., Wiseman, J., Chan, L.K.: Video as context and conduit for problem-based learning. In: Bridgesm, S., Chan, L., Hmelo-Silver, C. (eds.) Educational Technologies in Medical and Health Sciences Education, vol. 5, pp. 57–77. Springer, Cham (2016)

Hogaboam, P.T., Chen, Y., Hmelo-Silver, C.E., Lajoie, S.P., Bodnar, S., Kazemitabar, M., Chan, L.K.: Data dashboards to support facilitating online problem-based learning. Quart. Rev. Distance Edu. **17**(3), 75–91, 95–97 (2016)

Kazemitabar, M.A., Bodnar, S., Hogaboam, P., Chen, Y., Sarmiento, J.P., Lajoie, S.P., Chan, L., et al.: Creating instructor dashboards to foster collaborative learning in on-line medical problem-based learning situations. In: Zaphiris, P., Ioannou, A. (eds.) LCT 2016. LNCS, vol. 9753, pp. 36–47. Springer, Cham (2016)

Lajoie, S.P., Hmelo-Silver, C.E., Wiseman, J.G., Chan, L.K., Lu, J., Khurana, C., Kazemitabar, M.: Using online digital tools and video to support international problem-based learning. Interdiscip. J. Prob. Based Learn. **8**(2), 6 (2014)

A Framework to Recommend Appropriate Learning Materials from Stack Overflow Discussions

Ashesh Iqbal[1], Mohammad Shamsul Arefin[1(✉)],
and Mohammad Ali Akber Dewan[2]

[1] Computer Science and Engineering Department, Chittagong University of Engineering and Technology, Chittagong 4349, Bangladesh
iqbalashesh@gmail.com, sarefin@cuet.ac.bd
[2] Canada Athabasca University, Athabasca, Canada
adewan@athabascau.ca

Abstract. In this paper, we present a supervised machine learning based recommendation strategy that analyzes Stack Overflow posts to suggest informative sentences that is useful for programming tasks. We have conducted several experiments and found that our approach can successfully recommend useful information.

Keywords: Text classification · Supervised learning
Crowd knowledge · Recommendation systems

1 Introduction

Stack Overflow has gained the reputation of being a reliable forum where users get quick responses to their questions related to computer programming and that too with high level of accuracy. Researches [1–3] related to stack Overflow data have helped to understand the power of programming-specific Q&A forums and how far these forums are serving as learning platforms. Researchers have even deduced that the answers on Stack Overflow often become a substitute for official product documentation – when the official documentation is sparse or not yet existent [4]. Parnin et al. [2] claimed that Stack Overflow has grown into a tremendous repository of user-generated content that complements traditional technical documentations. In this paper we develop a supervised learning based tool that exploits the knowledge repository available from the Stack Overflow discussions to generate learning materials related to PHP and Python.

2 Methodology

Our proposed system consists of an interface module, a search module, a processor module and an input-output module. Stack Overflow makes its data publicly available [8] in Extensible Markup Language (XML) format under the Creative

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 446–449, 2018.
https://doi.org/10.1007/978-3-319-91464-0

Commons license. We downloaded a data dump containing a total of 3,34,56,633 posts, spanning from July 2008 to September 2016 and imported the XML files into MySQL relational database.

2.1 Training and Test Set Generation

Supervised learning requires labeled data. We selected from the constructed subset, a batch of 1,000 sentences and manually annotated them with a yes/no rating to indicate whether it was informative. We define "informative sentences" as the ones that are "meaningful on its own and conveys specific and useful information". During manual labelling, we followed a set of rules proposed by Treude et al. [7]. The training set was excluded from the dataset obtained from preprocessing and the remaining 7,18,614 sentences constituted the test set. In the next step, we generated the Attribute-Relation File Format (ARFF) files from the training and the test sets. This is the native format for the machine learning tool we used.

2.2 Feature Extraction

We defined 18 attributes to characterize our data. The construction of our feature set is based on careful inspection of our corpus. It is safe to say that our feature set captures the structural, syntactic and metadata information of the dataset. Out of these 18 attributes, three are related to the number of occurrences of keyword terms in the question body and answer body of a post, four are related to the presence or absence of source code in the question body and answer body and the remaining ones are either directly obtained or calculated from the post metadata.

2.3 Classification

To conduct supervised learning from our training dataset, we used the WEKA workbench, which is recognized as a landmark system in data mining and machine learning [9]. To remove the imbalance we applied the Synthetic Minority Oversampling Technique (SMOTE) and increased the number of "informative" instances by oversampling. We tested five different machine learning algorithms on our training set.

2.4 Ranking and Categorization of Result

During the classification operation, WEKA measured a level of confidence for each prediction made on the "never-before-seen" instances. We exploit this information to rank the "informative" sentences extracted from our test set. To devise a categorization rule in our framework, we followed the approaches used in [5, 6] on topic-modelling.

3 Experiments

To conduct evaluation of the classifier performance in our experiments we take both accuracy and f-measure as a performance metric. Table 1 shows the accuracy, precision, recall, and f-measure for the classifiers mentioned in the previous section.

Table 1. Performance of Different Classifiers

Classifier	Accuracy (%)	Precision	Recall	F–measure
Decision List	95.3093	0.953	0.953	0.953
Decision Tree	95.8763	0.959	0.959	0.959
k-NN	89.7938	0.901	0.898	0.898
Random Forest	98.3505	0.984	0.984	0.984
SVM	72.2165	0.723	0.722	0.722

4 Conclusion

Though the rise of social media has resulted in huge amount of information for programmers on the Internet, very often it can be difficult for a coder to determine where a particular piece of information is stored. In our work, we have presented an approach to leverage the Q&A crowd knowledge.

References

1. Barzilay, O., Treude, C., Zagalsky, A.: Facilitating crowd sourced software engineering via stack overflow. In: Sim, S.E., Gallardo-Valencia, R.E. (eds.) Finding Source Code on the Web for Remix and Reuse, pp. 289–308. Springer, New York (2013). https://doi.org/10.1007/978-1-4614-6596-6_15
2. Parnin, C., Treude, C., Grammel, L., Storey, M.-A.: Crowd documentation: Exploring the coverage and the dynamics of api discussions on stack overflow. Technical report, Georgia Institute of Technology (2012)
3. Joorabchi, A., English, M., Mahdi, A.E.: Text mining stackoverflow: an insight into challenges and subject-related difficulties faced by computer science learners. J. Enterp. Inf. Manage. **29**(2), 255–275 (2016)
4. Treude, C., Barzilay, O., Storey, M.-A.: How do programmers ask and answer questions on the web? Nier track. In: 33rd International Conference on Software Engineering (ICSE), pp. 804–807. IEEE (2011)
5. de Souza, L.B., Campos, E.C., Maia, M.D.A.: On the extraction of cookbooks for apis from the crowd knowledge. In: Software Engineering (SBES), 2014 Brazilian Symposium on, pp. 21–30. IEEE (2014)
6. Bajaj, K., Pattabiraman, K., Mesbah, A.: Mining questions asked by web developers. In: Proceedings of the 11th Working Conference on Mining Software Repositories, pp. 112–121. ACM (2014)

7. Treude C., Robillard, M.P.: Augmenting api documentation with insights from stack overflow. In: Proceedings of the 38th International Conference on Software Engineering, pp. 392–403. ACM (2016)
8. Stack Exchange Data Dump. https://archive.org/details/stackexchange/. Accessed 14 Oct 2016
9. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: an update. ACM SIGKDD Explor. Newsl. **11**(1), 10–18 (2009)

What Does It Mean to Provide the Right Level of Support During Tutorial Dialogue?

Sandra Katz[1](✉), Patricia Albacete[1], Irene-Angelica Chounta[2], Pamela Jordan[1], Dennis Lusetich[1], and Bruce M. McLaren[3]

[1] University of Pittsburgh, Pittsburgh, PA 15260, USA
katz@pitt.edu
[2] Institute of Education, University of Tartu, Tartu, Estonia
[3] Carnegie Mellon University, Pittsburgh, PA 15217, USA

Abstract. We describe and illustrate factors that specify what it means for a tutor to provide different "levels of support", based on our analyses of models of the levels of support provided during human tutoring and teacher-led small group work. We then show how we used these factors to implement contingent scaffolding in a tutorial dialogue system for physics.

Keywords: Natural-language tutoring systems · Scaffolding
Student modeling

1 Introduction

Studies of human tutoring and teacher guidance of small group work have shown that the extent to which support is contingent upon (i.e., tailored to) students' understanding and performance predicts achievement [e.g., 1–3]. These findings have prompted educators and educational psychologists to operationalize "contingent scaffolding" in order to effectively support students during classroom instruction, human tutoring, and interactions with an automated tutor in tutorial dialogue systems. Achieving this aim requires specifying what it means to provide the *right level of support* (LOS) to a student, at just the right time.

We addressed this question in the process of developing Rimac, a tutorial dialogue system designed to enhance students' conceptual understanding of physics [e.g., 4]. Rimac engages students in reflective dialogues after they have solved a physics problem on paper and have watched an annotated video of a correct solution. Rimac's dialogues are developed using an authoring framework called *Knowledge Construction Dialogues* (KCDs), which engage students in a series of carefully ordered questions known as a *Directed Line of Reasoning* (DLR) [5]. To our knowledge, Rimac is the only tutorial dialogue system that implements a student modeling engine that drives decisions about what content to address next during a dialogue and how to discuss focal content—that is, through which scaffolding strategies and at what level of support? These decisions depend on the student model's assessment of the student's understanding of the knowledge components associated with each step of a DLR.

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 450–453, 2018.
https://doi.org/10.1007/978-3-319-91464-0

In order to specify decision rules that can tailor the support provided at a particular step during a DLR to the student model's predictions, we examined prior research aimed at modeling the levels of support provided during tutoring and teacher-guided small group work [6]. This poster illustrates factors that operationalize "levels of support", shows how we incorporated these factors into rules to drive contingent scaffolding in Rimac, and describes an in-progress classroom study to evaluate the tutor.

2 Factors that Adjust Support in Questions and Feedback

Several frameworks have been developed to model the different levels of support provided in tutors' (and teachers') questions and feedback on students' responses. It is common for LOS framework developers to characterize their model in terms of broad dimensions like different "degrees of tutor control" and "degrees of cognitive complexity" [e.g., 2, 3, 7], such as the one posited by van de Pol et al. (2014) shown in Table 1. However, a closer look at the description of each level in a given framework revealed that tutor/teacher questions and feedback vary according to more specific factors, which can be incorporated within dialogue decision rules. For example, in van de Pol et al.'s LOS framework (Table 1), the "degree of teacher control" (TDc) depends on factors such as *response length* (e.g., yes/no or choice of options, versus elaborate response), *how much information the teacher provides* in a question or feedback, and a question's *level of abstraction*—for example, does the question provide a "hint or suggestive question" or more directive information?

Table 1. A Sample Level of Support Framework

TDc1 *Lowest control—teacher:* • Provides no new content • Elicits an elaborate response • Asks a broad and open question	**TDc4** *High control—teacher:* • Provides new content • Elicits a response • Gives a hint or suggestive question
TDc2 *Low control—teacher:* • Provides no new content • Elicits an elaborate response, mostly for an elaboration or explanation • Asks a more detailed but still open question	**TDc5** *Highest control—teacher:* • Provides new content • Elicits no response • Gives an explanation or the answer to a question
TDc3 *Medium control—teacher:* • Provides new content • Elicits a short response (yes/no or choice)	

Adapted from van de Pol (2012)
TDc = degree of teacher control

Given the quantitative nature of our domain, we further specified *level of abstraction* in terms of factors such as *whether to refer to variables in abstract terms or in terms of the problem* (e.g., "velocity" vs. "velocity of the bicycle"), *whether to provide the name of a law or an equation* (e.g., $F_{net} = m * a$, vs. Newton's Second

Law), and *whether to define the symbols in an equation* (e.g., v = velocity). We then used these factors to specify decision rules to adapt the tutor's support to students' knowledge level, according to their student model. For example, in Table 2, the rule for providing a high LOS (left column) would produce a question like, "Using Newton's Second Law (F_{net} = m * a) and knowing that the net force on the man in the elevator is zero, let me ask you about the man's acceleration. In which direction does the man's acceleration point"? In contrast, the rule for providing a low LOS (right column) would produce, "In which direction does the acceleration point?"

Table 2. Sample decision rules for question asking (differences *italicized*)

If the student's probability of answering the next question correctly is *low*:	**If the student's probability of answering the next question correctly is *high*:**
State quantities with reference to the problem Provide a hint or other type of support Provide the name of the law/definition in equation form Do not define symbols and/or variables Do not ask the question again if the response is incorrect	Reference quantities in *abstract terms* *Do not provide* a hint or other type of support *Do not provide* the name of the law or definition Do not define symbols and/or variables *Re-ask the question* if the response is incorrect

3 Conclusion

Our review of level of support frameworks revealed that broad dimensions such as "different degrees of tutor control" are too imprecise to guide the design of adaptive support in a tutorial dialogue system. We therefore dug deeper into these frameworks and uncovered factors that informed specification of decision rules to drive contingent scaffolding in Rimac. An in-progress evaluation of the tutor at several high schools in the Pittsburgh PA area, U.S.A., compares this dynamically updated, student model and decision rule-driven version of Rimac with a prior version that provides a static, less adaptive form of scaffolding based on students' pretest scores [4].

Acknowledgements. This research, which was supported by the Institute of Educational Sciences, U.S. Department of Education, through Grant R305A150155 to the University of Pittsburgh. The opinions expressed are those of the authors.

References

1. Wood, D., Middleton, D.: A study of assisted problem-solving. Br. J. Psychol. **66**(2), 181–191 (1975)
2. Pratt, M.W., Savoy-Levine, K.M.: Contingent tutoring of long-division skills in fourth and fifth graders: Experimental tests of some hypotheses about scaffolding. J. Appl. Dev. Psychol. **19**(2), 287–304 (1998)

3. van de Pol, J., Volman, M., Oort, F., Beishuizen, J.: Teacher scaffolding in small-group work: an intervention study. J. Learn. Sci. **23**(4), 600–650 (2014)
4. Jordan, P., Albacete, P., Katz, S.: Adapting step granularity in tutorial dialogue based on pretest scores. In: André, E., Baker, R., Hu, X., Rodrigo, M., Mercedes, T., du Boulay, B. (eds.) AIED 2017. LNCS, vol. 10331, pp. 137–148. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-61425-0_12
5. Evens, M., Michael, J.: One-on-One Tutoring by Humans and Computers. LEA Inc., Mahwah (2006)
6. Katz, S., Albacete, P., Jordan, P., Lusetich, D., Chounta, I-A., McLaren, B.M.: Operationalizing contingent tutoring in a natural-language dialogue system. In: Craig, S. (ed.) Tutoring and Intelligent Tutoring Systems. Nova Science Publishers, New York (in press)
7. Nathan, M.J., Kim, S.: Regulation of teacher elicitations in the mathematics classroom. Cogn. Instr. **27**(2), 91–120 (2009)

Building Student Models in a Non-scaffolded Testing Environment

H. Nguyen and C. W. Liew(✉)

Department of Computer Science, Lafayette College, Easton, PA 18042, USA
{nguyenha,liewc}@lafayette.edu

Keywords: Computer science · Data structures · Student modeling

1 Introduction

Balanced binary search tree is a domain where the inputs are graphical in nature. Conventional question formats such as multiple-choice would therefore constrain the student's answers and allow for the possibility of guessing. We present a tutoring system in this area, where the pre-tests and post-tests are designed to minimize scaffoldings and simulate a real paper exam. The pre-test answers are analyzed to construct a student model that represents the system's probabilistic understanding of student's errors [8]. We evaluate the accuracy of the model by using it to predict the students' performance in a subsequent tutoring session. Our results show that standard Bayesian models and techniques can still be effective in this non-scaffolded environment.

2 Red-Black Trees

A red-black tree is a self balancing binary search tree that has the following properties [4]:

1. The nodes of the tree are colored either red or black.
2. The root of the tree is always black.
3. A red node cannot have any red children.
4. Every path from the root to a null link contains the same number of black nodes.

The top-down algorithm to insert or delete a value from a red-black tree starts at the root and, at every iteration, moves down to the next node, which is a child of the current node. At each node, it applies one or more transformation rules so that when the actual insertion (or deletion) is performed no subsequent actions are needed to maintain the tree's properties. Other types of balanced trees also use a similar approach. In our work we used red-black tree as an exemplar to evaluate our ideas and implementations, but they should be applicable to balanced trees in general.

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 454–456, 2018.
https://doi.org/10.1007/978-3-319-91464-0

3 Student Modeling

Because of the non-scaffolded design of the test environments, we do not have any knowledge about which node the student is at or which transformation she is trying to perform. To extract relevant data, we have devised a grading algorithm [6] that, given the problem prompt (a starting tree and the number to insert/delete) and a sequence of trees that the student submitted, can determine if the student is correct. In case there are errors, the algorithm also identifies the location, type and context of the first error occurred.

Using this information, we build a two-part Bayesian network similar to that of the ANDES physics tutor [3]. The domain-general network encodes long-term knowledge and represents the system's assessment of the student's rule mastery after the last performed exercise. The task-specific network encodes the student's rule mastery in a specific exercise. Each tree transformation gets one representative Rule node, while the Context-Rule nodes are based on the error contexts identified by the grading algorithm. At the end of each problem, the task-specific network is discarded, but the probabilities of all Context-Rules are saved to the domain-general network, so that they can be used as priors for the next time these contexts appear. The mechanism to dynamically generate the network structure would allow each student to have an individualized model and the tutor's framework to be more applicable in other domains.

4 Evaluation and Results

We evaluated our algorithm for constructing student models on data from students in a computer science class at our institution. The data was taken from three semesters - Fall 2016 (29 students), Spring 2017 (50 students) and Fall 2017 (26 students). The pre and post tests are identical in content, both consisting of a small number of exercises in which students attempt to insert (delete) a node, given a starting tree. The accuracy of our model in predicting student performance is shown in Table 1.

When evaluating post-test answers, we identified error contexts in the same manner as we did in the pre-test. For each error context, we check whether that error has been identified in the student model before, and if it has, how confident the model is in predicting that the student does not make the same error again. Our results show that (1) there are no error contexts that have not been previously identified in the tutor, (2) after the tutoring session, the mastery probability of 70% of the error contexts are higher than 80%, and (3) in 91% of the times, if the mastery probability of a context is higher than 80%, the student does not make an error in that context in the post-test.

5 Conclusion

This paper describes an intelligent tutoring system whose assessment environments are designed to be consistent and without scaffolding. We have devised

Table 1. Student model's accuracy on the insertion tutor (top) and deletion tutor (bottom). The columns, from left to right, respectively refer to the followings: semester name, number of students, number of average and total correct predictions, mean accuracy, standard deviation of accuracy, lowest and highest accuracy across all students in the semester. The mean values are averaged over all students in each semester.

Semester	Mean.Correct/Total	Mean.Acc	Stdev.Acc	Min.Acc	Max.Acc
Fall 2016	268/372	72%	4%	63%	81%
Spring 2017	259/399	66%	8%	50%	86%
Fall 2017	267/371	72%	5%	62%	83%

Semester	Mean.Correct/Total	Mean.Acc	Stdev.Acc	Min.Acc	Max.Acc
Fall 2016	270/383	70%	5%	64%	82%
Spring 2017	268/383	70%	4%	61%	80%
Fall 2017	351/461	76%	4%	68%	83%

a framework to automatically construct a student model from pre-test answers, and to evaluate the tutor's effectiveness based on post-test results. Our results show that a student model built from non-scaffolded testing environments with less data but more accurate information, can effectively predict students' performance, with an average accuracy of 70%. As the next step, we would like to use the model's knowledge to generate dynamic and individualized exercises for each student in the tutoring session, thereby ensuring that the tutor can cover all of the errors that the student has encountered.

References

1. Baker, R., Pardos, Z., Gowda, S., Nooraei, B., Heffernan, N.: Ensembling predictions of student knowledge within intelligent tutoring systems. In: Adaption and Personalization, User Modeling, pp. 13–24 (2011)
2. Conati, C.: Bayesian Student Modeling. Advances in Intelligent Tutoring Systems, pp. 281–299 (2010)
3. Conati, C., Gertner, A., Vanlehn, K.: Using bayesian networks to manage uncertainty in student modeling. User Model. User-Adap. Inter. **12**(4), 371–417 (2002)
4. Cormen, T.H.: Introduction to Algorithms. MIT press (2009)
5. Kastner, M., Stangla, B.: Multiple choice and constructed response tests: do test format and scoring matter? Procedia-Soc. Behav. Sci. **12**, 263–273 (2011)
6. Liew, C.W., Nguyen, H.: Determining what the student understands - assessment in an unscaffolded environment. In: International Conference on Intelligent Tutoring Systems, Springer (2018)
7. Liew, C.W., Xhakaj, F.: Teaching a complex process: insertion in red black trees. In: Conati, C., Heffernan, N., Mitrovic, A., Verdejo, M.F. (eds.) AIED 2015. LNCS (LNAI), vol. 9112, pp. 698–701. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-19773-9_95
8. Pelánek, R.: Metrics for evaluation of student models. J. Educ. Data Min. **7**(2), 1–19 (2015)

Applying Human-Agent Team Concepts to the Design of Intelligent Team Tutoring Systems

Kaitlyn Ouverson, Mariangely Iglesias Pena, Jamiahus Walton, Stephen Gilbert, and Michael Dorneich

Iowa State University, Ames, USA
gilbert@iastate.edu

Abstract. Intelligent tutoring systems, having been relatively successful at emulating the results of human tutors for certain learning domains, are now being developed for intelligent team tutoring systems (ITTSs).With the addition of multiple humans in the system, modeling the communication and coordination between the humans and the tutoring agent grows combinatorically and presents significant challenges to ITTS development. Answers to some of these challenges can be found in the research field of human-agent teamwork. This paper applies common concepts in human-agent team literature (such as task allocation, adaptive automation triggers, and behavior modeling), to steps used to author a team tutor.This research should enable developers of ITTSs to draw more efficiently on research from two otherwise separate research areas.

Keywords: Intelligent team tutoring systems · Intelligent tutoring systems
Intelligent team training systems · Human-agent teamwork
Human-agent collaboration · Adaptive automation

1 Introduction

An intelligent team tutoring system (ITTS) is one that focuses on improving teamwork skills in addition to improving taskwork skills. Ten steps for authoring an ITTS are described in [1]. While traditional research on the development of intelligent tutoring systems has drawn heavily on the fields of psychology, computer science, and learning sciences, it has drawn less on the engineering fields of human-machine systems and adaptive automation. Fields such as Computer Supported Collaborative Learning (CSCL) or Work (CSCW) can also contribute, but this short paper focused on bridging work that is typically more CS/psychologically based with the human-agent teaming constructs from human factors engineering.

When designing a team tutor, each component of an ITS (task, learner, domain, and feedback models) grows more complex [2]. As demonstrated by previous ITTS research, feedback additionally requires the consideration of many different components, including: the recipient of feedback (individuals vs. the entire team), feedback method (audio, visual, etc.), feedback content, feedback timing (just-in-time vs after action), and level of privacy (public vs private) [3].

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 457–459, 2018.
https://doi.org/10.1007/978-3-319-91464-0

While an agent can be authored to take on any role in a team, this effort to apply HAT theory to ITTSs will focus on the agent in a supervisory or facilitative role – the typical roles of a human tutor.

2 Mapping HAT to ITTS

This research mapped several conceptual constructs from HAT to ITTS development to inform future ITTS authors.

Task allocation consists of determining which roles are necessary for task completion and assigning those roles to humans or autonomous agents [4]. Analogously, if a tutor is to train any learner on a task or set of tasks, it must be able to distinguish those tasks at the correct granularity to deliver proper feedback on students' performance [1].

Adaptive automation systems model HAT interactions using a Perceive, Select, Act paradigm. The system first perceives the state of the world (*e.g.*, system state, task state, human state), selects any appropriate adaptations that are triggered (e.g., change in task allocation, change in information shown to participants), and then acts (implements the adaptation) via changes to the automation of changes to the user interface [5]. Such adaptive triggers are similar in function to the behavioral markers specified in an ITTS's feedback conditions [6], as they are used to trigger personalized responses such as adaptive feedback.

Level of Automation (LoA) refers to the degree of autonomy that an agent has. At least 12 different frameworks for LoA have been described (see [7] for a comparison), but their common goal is to delineate the extent to which the human vs. the automated agent makes decisions and is responsible for a task. As ITTSs are developed with tutors that play partnership roles with the learners rather than authority roles, this construct will be key for describing the authority relationships and task responsibilities within the learner-agent relationships.

Similar to the concept of error resistance discussed in human-agent teamwork literature, ***resilience engineering*** is a human factors engineering effort to reduce system failure by modeling the ways in which organizations, systems, and people adapt to problems with resilience and prevent disruption [8]. An ITTS skill model is more complex than its single-learner counterpart, modeling both individual team member skills and the team's skills, as well as modeling task-related skills and teaming-related skills at both individual and team levels. Rather than attempting to model all of these complex dynamics, the authors suggest that a resilience engineering approach may be simpler, especially in the area of team skills, by focusing on monitoring known risks to team dynamics.

The construct of ***etiquette*** arises from previous research describing an individual's interactivity expectations of a software agent, often based on the perception that agent interactions should be modeled on or mirror human social constructs [9, 10]. Research on feedback within human-agent systems has shown how different forms of adaptive feedback, both in terms of the medium – how they are presented to the user (e.g., visually vs. by audio, just-in-time or an after-action debrief) – and the message (content) influence a user's ability to improve performance in a collaborative task [11].

Behavior modeling has roots in several fields, e.g., ACT-R efforts from cognitive science [12], likely more familiar to ITS researchers, as well as HAT models stemming from systems engineering and control theory that using an engineering lens to model and simulate interactions between operators and their systems, e.g. [13].

These constructs from the HAT domain can be mapped onto the 10 steps of authoring an ITTS as described in [1], and this mapping should create a more robust team tutor, both in terms of its model of its learners' actions and its own interactions with learners.

References

1. Gilbert, S.B., et al.: Creating a team tutor using GIFT. Int. J. Artif. Intell. Educ. (2017)
2. Sottilare, R.A., Holden, H., Brawner, K., Goldberg, B.: Challenges and emerging concepts in the development of adaptive, computer-based tutoring systems for team training. U.S. Army Research Laboratory – Human Research and Engineering Directorate. Orlando, Florida (2011)
3. Walton, J., et al.: Modality and Timing of Team Feedback: Implications for GIFT. In: Proceedings of the 2nd Annual Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium. Army Research Laboratory (2014)
4. Lee, J., Wickens, C., Liu, Y., Boyle, L.N.: Designing for People: An Introduction to Human Factors Engineering, 3rd edn. CreateSpace, Charleston (2017)
5. Feigh, K.M., Dorneich, M.C., Hayes, C.C.: Toward a characterization of adaptive systems: a framework for researchers and system designers. Hum. Factors: J. Hum. Factors Ergon. Soc. **54**, 1008–1024 (2012)
6. Sottilare, R.A., et al.: Designing adaptive instruction for teams: a meta-analysis. Int. J. Artif. Intell. Educ. 1–40 (2017)
7. Vagia, M., Transeth, A.A., Fjerdingen, S.A.: A literature review on the levels of automation during the years. What are the different taxonomies that have been proposed? Appl. Ergon. **53**, 190–202 (2016)
8. Madni, A.M., Jackson, S.: Towards a conceptual framework for resilience engineering. IEEE Syst. J. **3**, 181–191 (2009)
9. Hayes, C.C., Miller, C.A.: Human-Computer Etiquette: Cultural Expectations and the Design Implications They Place on Computers and Technology, 1st edn. Auerbach, Boston (2010)
10. Parasuraman, R., Miller, C.A.: Trust and etiquette in high-criticality automated systems. Commun. ACM **47**, 51–55 (2004)
11. Yang, E., Dorneich, M.C.: Evaluation of Etiquette Strategies to Adapt Feedback in Affect-Aware Tutoring. In: Proceedings of the Human Factors and Ergonomics Society (HFES) 2016 Annual Meeting, vol. 60, 393–397 (2016)
12. Anderson, J.R.: ACT: a simple theory of complex cognition. Am. Psychol. **51**, 355 (1996)
13. Pentland, A., Liu, A.: Modeling and prediction of human behavior. Neural Comput. **11**, 229–242 (1999)

Concept-Based Learning in Blended Environments Using Intelligent Tutoring Systems

Ines Šarić[1](✉), Ani Grubišić[1], Slavomir Stankov[2], and Timothy J. Robinson[3]

[1] Faculty of Science, University of Split, Split, Croatia
{ines.saric,ani.0grubisic}@pmfst.hr
[2] Retired Full Professor, Split, Croatia
slavomirstankov@gmail.com
[3] Department of Statistics, University of Wyoming, Laramie, USA
tjrobin@uwyo.edu

Abstract. CoLaB Tutor and AC-ware Tutor are Intelligent Tutoring Systems (ITSs) that are based on concept-based learning and are notable due to the fact they are relatively easy to generalize to multiple knowledge domains. In this research study we investigate the performance of CoLaB Tutor, AC-ware Tutor, and Moodle in a blended learning environment for an introductory computer programming course. In our study, regular face-to-face lectures and laboratory exercises were complemented with online learning at the students' own pace, time and location. Our study revealed that CoLaB Tutor students had moderately higher knowledge gains than those students in the AC-ware and Moodle groups. The prediction of student success (pass/fail) for a basic knowledge post-test revealed an overall classification rate of 73,5% for the CoLaB Tutor group (completed knowledge and online score as predictors), 71,4% for the AC-ware group (completed knowledge as predictor) and 70% for the Moodle group (time spent online as predictor). Additionally, students that used ITSs on average passed through more knowledge online than students that used LMS, while students that used LMS on average spent more time online.

Keywords: Experimental studies · Intelligent tutoring systems
Blended learning environments · Conceptual knowledge

1 Introduction

Compared to Learning Management Systems (LMS), such as the widely used Moodle (moodle.org) and Blackboard (blackboard.com) systems, ITS platforms generally require higher development costs and are often utilized only for specific knowledge domains. Controlled Language Based Tutor (CoLaB Tutor) and Adaptive Courseware Tutor Model (AC-ware Tutor) are ITSs that share the characteristic of ontological domain knowledge representation [1]. These platforms are particularly appealing due to the fact that they are easily generalizable to multiple knowledge domains. In this research, we investigate the performance of CoLaB Tutor, AC-ware Tutor, and

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 460–462, 2018.
https://doi.org/10.1007/978-3-319-91464-0

Moodle, when utilized within a blended learning environment for an introductory computer programming course.

2 Research Methodology, Results and Findings

The research study included 187 undergraduate students from the Faculty of Science at the University of Split. Along with traditional face-to-face lectures and laboratory exercises that were held for 4 h per week, online instruction occurred at the students' own pace, time and location. The type of blended learning in which students learn conceptual knowledge before coming to class is called - a flipped model of blended learning environment. The purpose of the pre-class learning was to have students better prepared for the class and thus allowing the teacher to spend face-to-face class for clarifying and applying the conceptual knowledge. One week before the experiment, the entire class was introduced to the notion of concept mapping technique. A week after the introductory concept mapping lecture, a pre-test was given to students for an assessment of baseline knowledge of computer programming. Using the pre-test scores, students were assigned into 3 experimental groups for online instruction. Two types of post-tests were used: the basic computer programming knowledge test which was also used as a pre-test, and a computer programming skill-based post-test.

The aim of this research is to address the following research questions:

(i) Are there statistically significant differences in CoLaB, AC-ware Tutor, and Moodle in terms of pre-test and post-test scores (i.e. gain in knowledge)?
(ii) What is the relationship between students' online behavior on CoLaB Tutor, AC-ware Tutor and Moodle, and knowledge performance on the post-tests?
(iii) What is the student experience when using CoLaB Tutor, AC-ware Tutor and Moodle in the learning process?

A Kruskal Wallis test was used to compare the pre-test scores across groups and no significant difference was observed (p = 0,732). By comparing the groups in terms of test score differences, the Kruskal Wallis test revealed a moderate difference (p = 0,097) with the CoLaB Tutor group having slightly higher gains in test scores. Based on 95% bootstrapped confidence intervals on the median test score gain for each of the three groups, the median test score gain was 27 points for the CoLaB Tutor group, 23.5 points for the Moodle group and 20 points for the AC-ware group.

Groups were also compared using both sets of raw post-test scores. The 95% bootstrapped confidence intervals on the median basic knowledge post-test scores and for the skill-based post-test scores reveal no statistical difference in the groups but we do observe slightly higher scores for the students in the CoLaB Tutor group.

Online learning behavior was measured using Knowledge Tracking Variables (KTV) for online learning behavior [2]. Knowledge tracking variables include: (i) the total number of concepts (completed knowledge) (#Knowledge), (ii) the total online quiz score (#Score), (iii) the total time spent online (#Time), and (iv) the total number of student logins (#Logins). In the CoLaB Tutor group, 67.32% of the students were successful on the basic knowledge post-test, 64.25% of the AC-Ware group students and 63.4% of the Moodle group were successful on the same post-test. The final logistic regression model for each group was determined using AIC. Note that #Concepts was an important predictor in both ITS groups while 'Time' was the only important predictor in the Moodle group.

The predictions of student success (pass/fail) for the basic knowledge post-test revealed an overall classification rate of 73,5% for CoLaB Tutor group (completed knowledge and online score as predictors), 71,4% classification rate for AC-ware Tutor group (completed knowledge as predictor) and 70% classification rate for Moodle group (time spent online as predictor). The predictions of student success (pass/fail) for the skill-based post-test revealed the overall classification rate of 73,7% for CoLaB Tutor group (all KTVs as predictors), and 62,5% classification rate for AC-ware Tutor group (online score as predictor). No KTVs were statistically significant in the model for predicting student success in the Moodle group.

The descriptive statistics of online learning behavior revealed that CoLaB Tutor and AC-ware Tutor students passed over 96% of the conceptual knowledge aimed for online learning. Moodle students on the other hand completed only 47% of the online lessons. In terms of the total time and the number of logins on each e-learning platform, Moodle students logged-in more times and spent more time online than the ITS student groups. The AC-ware group students tended to spend the least amount of time online – approximately 79 min spread across an average of 5 log-ins.

It is interesting to note that 70% of the CoLaB group students found the use of concept maps during experimental learning helpful, while 'only' 46% of Moodle student group and 35% of AC-ware student group found concept mapping helpful. Regarding the use of concept maps in future courses, 45% of the CoLaB student group said they will use this type of tool in the future while only 36% of the Moodle student group and only 25% of AC-ware student group indicated interest in using concept maps in the future.

Acknowledgements. This paper is part of the Adaptive Courseware & Natural Language Tutor project that is supported by the Office of Naval Research Grant No. N00014-15-1-2789.

References

1. Grubišić, A., et al.: Empirical evaluation of intelligent tutoring systems with ontological domain knowledge representation: a case study with online courses in higher education. In: Proceedings of the 13th International Conference Intelligent Tutoring Systems, ITS 2016, pp. 469–470. Zagreb, Croatia (2016)
2. Grubišić, A., et al.: Knowledge tracking variables in intelligent tutoring systems. In: Proceedings of the 9th International Conference on Computer Supported Education, CSEDU 2017, vol. 1, pp. 513–518. Porto, Portugal (2017)

Diagnosing Reading Deficiencies of Adults with Low Literacy Skills in an Intelligent Tutoring System

Genghu Shi[1,2(✉)], Andrew J. Hampton[1], Su Chen[1], Ying Fang[1], and Arthur C. Graesser[1]

[1] University of Memphis, Memphis, TN 38111, USA
gshi@memphis.edu
[2] Institute for Intelligent Systems, Memphis, USA

Abstract. We developed a version of AutoTutor that helps struggling adult learners improve their comprehension strategies through conversational agents. We hypothesized that the accuracy and time to answer questions during the conversation could be diagnostic of their mastery of different reading comprehension components: words, textbase, situation model, and rhetorical structure. The results show that adults' performance on more basic reading components (i.e., meaning of words) was higher than on the deeper discourse levels. In contrast, time did not vary significantly among the theoretical levels. The results suggested that adults with low literacy had higher mastery on basic reading levels than deeper discourse levels. The tracking of performance on the four theoretical levels can provide a more nuanced diagnosis of reading problems than a single overall performance score and ultimately improve the adaptivity of an ITS like AutoTutor.

Keywords: CSAL AutoTutor · Reading strategies
Comprehension framework

1 Introduction

We developed a version of a web-based intelligent tutoring system (AutoTutor) for adults with low literacy skills to improve their reading comprehension strategies in the Center for the Study of Adult Literacy (CSAL). AutoTutor for CSAL has 35 lessons that focus on distinct theoretical levels of reading comprehension articulated by Graesser and McNamara [1]. For each lesson, the system starts out assigning words or texts at a medium difficulty level and then asks 8 to 12 multi-choice questions about them. In this study, we tracked four theoretical levels (of the six defined in [1]). *Word* represents the lower-level basic reading components. The other three theoretical levels (*textbase, situation model*, and *rhetorical structure*) represent deeper discourse levels. We hypothesized that the accuracy and time on questions in AutoTutor could be diagnostic of adults' mastery of comprehension components. Therefore, by comparing the accuracy and time on questions of four theoretical levels, we can detect adults' strengths and weaknesses in reading competencies.

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 463–465, 2018.
https://doi.org/10.1007/978-3-319-91464-0

2 Methods

2.1 Participants

The participants were 52 adults recruited from CSAL literacy classes in Atlanta and Toronto. They completed a 100-hour intervention over four months. Their ages ranged from 16–69 years (Mean = 40, SD = 14.97). The majority of the participants were female (73.1%). All participants read at 3.0–7.9 grade levels.

3 Measures and Data Analysis

We extracted the adults' initial responses on medium level questions in each of the 29 lessons that focused on the four theoretical levels. All adults answered these initial medium questions before adaptively branching to easy or difficult questions in AutoTutor. The initial responses included accuracy (1 or 0) and time to select an answer (in seconds).

We performed a descriptive analysis by exploring the means and standard deviations of accuracy and time on questions of the four theoretical levels. Then we performed mixed effect models [2] on the two measures to test the difference among the four theoretical levels, with *question* as the unit of analysis. The random effects were participants, lessons, and questions; the fixed effect was theoretical level. Participants' random slopes on different theoretical levels and random intercepts of the interaction between lesson and question were also included in the models.

4 Results

Table 1 shows the means of accuracy and time on questions separately as a function of the four theoretical levels. The pattern of scores indicate that performance is highest and answer times are shortest for the *word* level (reference level in the analysis) compared to the three discourse levels (*textbase, situation model,* and *rhetorical structure*).

Table 1. Means and standard deviations of accuracies and time

		Word	Textbase	Situation model	Rhetorical structure
	No. of questions	1455	1981	5049	5071
Accuracy	Mean (SD)	0.80 (0.40)	0.69 (0.46)	0.67 (0.47)	0.69 (0.46)
Time	Mean (SD)	31.7 (30.4)	35.1 (30.2)	35.2 (31.6)	37.1 (38.1)

A Type II Wald Chi-square test on the logistic mixed effect model showed that accuracies were significantly different ($\chi^2(3) = 8.34$, $p = 0.04$) among the four theoretical levels. A post-hoc analysis with pairwise comparison showed only *word* pairs were significantly different. An ANOVA of type III with Satterthwaite on linear mixed effect model showed that time did not vary among the four theoretical levels, $F(3,25.8) = 0.058$, $p = 0.981$.

5 Discussion and Conclusion

The logistic mixed effect model indicates that adults' performance on *word* level was higher than the three discourse levels. This likely occurred because word items focused on individual words or single sentences which require low loads on working memory, whereas solving the items of deeper discourse levels is time-consuming, strategic, and taxing on cognitive resources. The time that adults spent on questions were not significantly different across theoretical levels, although times trended slower as theoretical levels progressed.

This study provides a more nuanced diagnosis of adults' reading problems within a multilevel reading comprehension framework than a single overall performance score could contribute. Future research should focus on designing standard reading tests and establishing norms for adult populations based on the multilevel framework that affords this diagnostically useful differentiation. Combining the testing results and the norm, researchers could develop more adaptive intelligent tutoring systems which provide customized learning contents to low literacy adults.

Acknowledgements. This research was supported by the National Center of Education Research (NCER) in the Institute of Education Sciences (IES) (R305C120001) and the National Science Foundation Data Infrastructure Building Blocks program under Grant No. (ACI-1443068).

References

1. Graesser, A.C., McNamara, D.S.: Computational analyses of multilevel discourse comprehension. Top. Cogn. Sci. **3**(2), 371–398 (2011)
2. Bates, D., Mächler, M., Bolker, B., Walker, S.: Fitting linear mixed-effects models using lme4. J. Stat. Soft. **67**, 1–48 (2015)

Everycoding: Combination of ITS and (M)OOC for Programming Education

Dongeun Sun and Hyeoncheol Kim(✉)

Korea University, Seoul, South Korea
{sunde41,hkim64}@gmail.com

Abstract. Both MOOC and ITS has its respective advantages in programming learning. As MOOC and ITS are complementary to each other, their integration will increase learning effectiveness. We developed the system ‘Everycoding’, which integrates MOOC and ITS to evaluate the effectiveness. We introduced two models in the system: programming knowledge model and reusable student model. Programming knowledge model represents programming concepts and encodes various types of learning contents in MOOC. Reusable student model is a student model that can be used for other courses in MOOC. In this paper, we present the models in the Everycoding.

Keywords: Programming tutoring · ITS · MOOC

1 Introduction

Teaching programming skills to students is not easy because every student has his/her own learning pace, learning style, knowledge level and preferred pedagogical type. Therefore, it is recommended for novice students to take advantage of 1:1 tutoring, peer collaborative learning or self-paced/motivated learning resources. As the number of learners who want to learn programming is increasing fast recently, it is highly required to introduce automated learning systems. Two most popular systems for programming education are ITS (intelligent tutoring system) and MOOC (Massive Open Online Course). ITS is able to support students with customized contents, feedback and evaluation. However, learning programming makes countless amounts of different situations where the ITS cannot guide all of them. MOOC supports various of instructions including video lectures, reading with conceptual questions, discussion boards, and various forms of learning by doing with peer feedback. If they are combined together, learning programming will be much more effective. Different from other researchers [1–3] suggesting architectural integration or blueprints for integration, we developed a working system ‘Everycoding’ and propose two functional models that combine the benefits from ITS and MOOC respectively.

This research was supported by the Korean MSIT (Ministry of Science and ICT), under the National Program for Excellence in SW (2015-0-00936) supervised by the IITP (Institute for Information & communications Technology Promotion)

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 466–469, 2018.
https://doi.org/10.1007/978-3-319-91464-0

2 Two Models Used in Everycoding System

Everycoding is both an Intelligent Programming Tutoring System (IPTS) and Programming Open Online Course system (OOC), that we developed. It is composed of knowledge model, student model and tutor model just as in ITS. In case of existing IPTS, reuse of the student model and the encoding of various kinds of domain knowledge were not big issues, because it did not support various kinds of learning as in MOOC. However, in the ITS integrated MOOC, different types of domain knowledge model and student model are needed.

2.1 Programming Knowledge Model

Each learning content in (M)OOC is encoded by Definition 1. The code is composed two elements: an identifier representing language type and content type, and a concept representing 9 different knowledge type. Each knowledge type is assigned with a value of understanding level of students who have completed learning of the content. The value is normalized according to the degree of difficulty. Example 1 is one example of learning content and will be encoded as {{Python, Code}, [(4,3), (3,2), (2,2), (2,1), (3,1), (1,1), (5,7), (1,1), (2,1)]}. The example says that the content is Python Code and contains operation concept of level 3 and degree of difficulty of 2. As no concepts of data structure and class is present in the code, both are assigned (1,1) because default value is 1.

Definition 1. Programming knowledge model.

Programming Knowledge ::= {Identifier | Programming Concept = [Basic, Operation, Variable, Input/Output, Data Structure, Condition, Iterate, Function, Class] }

Identifier ::= {[C |C++ |Java | Python] | [Code | video lectures | reading with concept | question/answer | peer feedback(comment)]}

Example 1. Python Code

```
x = int(input("Please enter an integer: "))
if x < 0:
… x=0
… print('Negative changed to zero')
elif x == 0:
… print('Zero')
else:
… print('More')
```

2.2 Reusable Student Model

We build a student model as in Definition 2. The two elements in the code are an identifier representing language type and a programming concept same as in Definition 1. The programming concept level for each student is available to be used in other content learning in the MOOC later on.

Definition 2. Reusable student model.

Programming Concept Accomplishment::={Identifier | Programming Concept}

Example 2 Examples of student model values of a student

[{{Python }, [(4,3), (2,2), (2,1), (3,1), (1,1), (5,7), (1,1) , (1,1) , (1,1)]}
{{C},[(4,1), (2,1), (2,1), (3,1), (1,1), (5,1), (1,1) , (1,1) , (1,0)]}]

The first example says that a student has knowledge of Python with 9 achievement values for 9 different concept types (Basic, Operation, Variable, Input/Output, Data Structure, Condition, Iterate, Function, Class). The value 1 means that the concept of Data Structure, Iterate, Function, Class is not learned yet. Value 0 means that there is no concept in the language type, for example, as (1,0) in the second example means that there is no concept of ‘Class’ in C language.

3 Conclusion

We introduced two functional models to combine ITS and MOOC systems for programming education. Each content is encoded with meta data including knowledge units and level of difficulty, and each student is encoded with knowledge levels of different programming concepts. The codes can be reused when the student tries to learn different languages and contents to provide them customized feedbacks and recommendations.

References

1. Aleven, V., Sewall, J., Popescu, O., Ringenberg, M., van Velsen, M., Demi, S.: Embedding intelligent tutoring systems in MOOCs and e-learning platforms. In: Micarelli, A., Stamper, J., Panourgia, K. (eds.) ITS 2016. LNCS, vol. 9684, pp. 409–415. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39583-8_49
2. Aleven, V., Sewall, J., Popescu, O., Xhakaj, F., Chand, D., Baker, R., Wang, Y., Siemens, G., Rosé, C., Gasevic, D.: The beginning of a beautiful friendship? intelligent tutoring systems and MOOCs. In: Conati, C., Heffernan, N., Mitrovic, A., Verdejo, M. (eds.) AIED 2015.

LNAI, vol. 9112, pp. 525–528. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-19773-9_53

3. Melis, E., Andrès, E., Büdenbender, J., Frischauf, A., Goguadze, G., Libbrecht, P., Pol-let, M., Ullrich, C.: ActiveMath: A Generic and adaptive web-based learning environment. Int. J. Artif. Intell. Educ. **12**(4) 385–407 (2001)

Adaptive Virtual Tutor Based on the Inference of the Student's Memory Content

Joanna Taoum(✉), Elisabetta Bevacqua, and Ronan Querrec

Lab-STICC, UMR 6285, CNRS, ENIB, Plouzane, France
{taoum,bevacqua,querrec}@enib.fr

Abstract. This research work presents an adaptive and embodied virtual tutor. Our proposed tutor is able to adapt the execution of a pedagogical scenario according to the estimated student's level of knowledge. To achieve such a goal, we rely on Mascaret, a meta-model for knowledge representation in a virtual environment and on an inference of the student's memory content. This inference permits the tutor to adapt the execution of the pedagogical scenario and to choose an individualized assistance according to the evolution of the student.

Keywords: Procedural learning · Adaptivity · Student's memory

1 Introduction

Virtual Reality is considered as one of the technologies with the most potential to improve procedural learning. However, procedures are learned gradually as a result of practice, for that, learners must repeat them. Throughout the repetitions, the tutor's pedagogical actions are usually scheduled using pedagogical scenarios. Taking into consideration that each student evolves differently during the repetitions, it is important to adapt the pedagogical scenario according to the student's evolution. The real-time adaptation of the pedagogical situation to a student is one of the major objectives of Intelligent Tutoring Systems (ITSs). Our proposed model[1] permits a tutor to execute a pedagogical scenario and especially to adapt its execution to the individual evolution of the student.

2 The MEMORA Model

Our work proposes a model to formalize the four ITS components (domain model, student model, tutoring model and interface) and the interactions between them.

Domain Model For the definition of the domain model, we use Mascaret [1]. It is a virtual reality meta-model based on the Unified Modeling Language (UML). It covers all the aspects of virtual environments semantic representation:

[1] This research work is partially supported by the Brittany Region.

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 470–472, 2018.
https://doi.org/10.1007/978-3-319-91464-0

domain's ontology, environment's structure, entities' behavior and both user's and agents' interactions and activities. In Mascaret, pedagogy is considered as a specific domain model. Pedagogical scenarios are implemented through UML activity diagrams containing a sequence of actions. These actions can be either pedagogical actions, like explaining a resource, or domain actions, like manipulating an object. The fact that Mascaret is a meta-model has two main interests. Firstly, specific domains are considered as data. This allows domain experts to provide the knowledge themselves in the ITS. Secondly, these data remain explicit during the simulation, thus they can serve as a knowledge base for agents.

Interface One of our main contributions to the interface model consists in embodying the virtual tutor. To achieve such a goal, we integrated the Embodied Conversational Agent (ECA) platform, Greta [2]. Greta characters are able to select and perform multi-modal communicative and expressive behaviors in order to interact naturally with the user. It is important to notice that in Mascaret, any entity which acts on the environment is considered as an *agent*. Particularly, the ECA and the human user are considered as *embodied* agents. The basic actions that an embodied agent is able to perform are: verbal communication (e.g. giving an information, answering), action realization (e.g. facial expression and actions that modify the environment) and navigation. The system is able to recognize the realization of each of these actions performed by the user.

Student Model In ITS, student models infer the student's cognitive and affective knowledge, to represent their relevant characteristics and the past interactions with the system [3]. As we are dealing with teaching human activities in industrial systems, the cognitive knowledge that our student model has to infer is related to memorization. This implies the transformation of stimuli (coming from the tutor and the environment) into knowledge that can be stored in memory. In our student model, we rely on the general theoretical framework proposed by Atkinson and Shiffrin [4] which divides human memory into three structural components (see Fig. 1): Sensory Memory (SM), Working Memory (WM) and Long-Term Memory (LTM). We propose an implementation of this framework

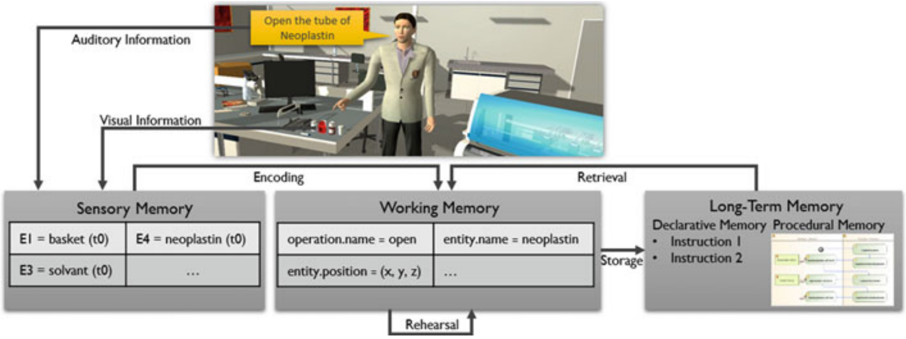


Fig. 1. Formalization of the encoding and structuring of instructions in the memory.

for learning procedures on industrial systems. In our work, incoming stimuli from the virtual environment and tutor are restricted to those related to vision and hearing. Thus, the student can see 3D objects and hear instructions uttered by the tutor about activities to realize. Therefore, we encode data about the objects and activities. For the formalization of the information encoding in all memories, we rely on the data formalism proposed by MASCARET. This formalism is hierarchical, which permits us to infer the knowledge level of the learner.

The role of the SM is to select relevant information among the continuous flow of stimuli that our senses deliver us. Perceived information is converted into a construct that can be stored in the SM. Only prominent information (e.g. objects that have been highlighted) is transferred from the SM to the WM. The WM stores and manipulates information based on the content of the SM and the LTM (prior knowledge). The level of complexity of the information that will be stored in the WM depends on the student's prior knowledge. By complexity of information we mean the type of formal representation and the number of attributes set. This prior knowledge is retrieved from the LTM. The transfer of some elements related to an action, from the WM to the LTM, takes place when the student completes the action [5]. The LTM is used to store permanently relevant information coming from the WM. It is composed of procedural memory (the procedure to learn) and declarative memory (domain model concepts). The choice of the information, its level of complexity and when it will be stored in the LTM depends on the pedagogical actions done by the tutor.

Tutor Behavior The goal of our proposed tutor behavior is to adapt the execution of the pedagogical scenario according to the student model represented in our work by the student's memory. The tutor behavior takes into account the action done by the student and compares it to the domain knowledge. If the realized action is expected by the tutor (e.g. correct action, right answer), then the transfer to the LTM occurs. The adaptation of the execution of the scenario takes place when the action performed by the student is unexpected (e.g. incorrect action, negative facial expression). In this case the tutor modifies the inference on the content of the WM and realizes another pedagogical action.

References

1. Chevaillier, P., Trinh, T., Barange, M., Devillers, F., Soler, J., De Loor, P., Querrec, R.: Semantic modeling of virtual environments using Mascaret. In: Proceedings of the 4th Workshop SEARIS, IEEE VR, Singapore (2001)
2. Niewiadomski, R., Bevacqua, E., Mancini, M., Pelachaud, C.: Greta: an interactive expressive ECA system. In: 8th International Conference AAMAS, pp. 1399–1400 (2009)
3. Nkambou, R., Mizoguchi, R., Bourdeau, J. (eds.): Advances in Intelligent Tutoring Systems, vol. 308. Springer, Berlin (2010)
4. Atkinson, R.C., Shiffrin, R.M.: Human memory: a proposed system and its control processes. In: Spence, K.W., Spence J.T. (eds.) The Psychology of Learning and Motivation: Advances in Research and Theory, vol. 2, pp. 89–105. (1968)
5. Ganier, F.: Factors affecting the processing of procedural instructions: implications for document design. IEEE Trans. Prof. Commun. **47**, 15–26 (2004)

Preliminary Evaluation of a Serious Game for Socio-Moral Reasoning

Ange Tato[1(✉)], Aude Dufresne[2], Roger Nkambou[1],
Frédérick Morasse[2], and Miriam H. Beauchamp[2]

[1] Université du Québec à Montréal, Montréal, Canada
angetato@gmail.com
[2] Université de Montréal, Montréal, Canada

Abstract. This paper presents the evaluation of a serious game that supports socio-moral reasoning assessment and learning. The game places learners in a 3D environment in which they face social dilemmas and are asked to provide and justify their opinion. The game includes Non-Player Characters (NPC) as friends who present their own opinions and social choices that reflect different levels of socio-moral reasoning (SMR) maturity. Usability was assessed via subjective measures (questionnaires) and the learning potential of the game was evaluated through a comparison of pre- and post-test assessment of the players' levels of SMR maturity. Results suggest that the game was appreciated by the players in terms of immersion and playability. Preliminary evaluation suggests that the game may also lead to improved SMR maturity.

Keywords: Moral reasoning · Social skills · Serious game · Learner model
Assessment · Social immersion

An Adapted Serious Game for SMR Development

The Socio-Moral Reasoning Aptitude Level (So-Moral) task [1, 2] is a computer measure in which children and adolescents are presented with visual social dilemmas representative of everyday life and asked to determine how they would react and justify their decisions. In the original task, expert coders are used to score the maturity of the verbatim justifications provided using a cognitive-developmental approach. Subsequently, an automated data mining model based on supervised text classification was developed using a large dataset of verbatims to assess individuals' SMR maturity automatically [3].

Considering the knowledge domain, it was important to introduce a social dimension inside the game in order to make the game more immersive and closer to the reality of the conditions in which adolescents would have to make similar decisions in daily life. To this end, Non-Player Characters (NPC) were integrated in the game, such that the main player was surrounded by them when he was presented the dilemmas. The player was asked what he would do when faced with socio-moral conflicts and then prompted to ask the NPC what they would do and to assess their opinions. Each NPC was assigned a SMR maturity level and their opinions were taken from verbatims of previous experimentations, which were assessed for that level and that

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 473–475, 2018.
https://doi.org/10.1007/978-3-319-91464-0

dilemma. Audio recordings by young actors were used to present the NPCs' answers and make interactions more realistic.

During the game, nine dilemmas were presented for which players were asked what they would choose to do and why. Their answers were recorded, and the verbal justification was transcribed using google speech to text API and analyzed using a data mining algorithm. The modeling of the socio-moral assessment was developed using a convolution neural network ([3]), with the previous responses manually classified by experts as the reference set (691 verbatims). For all the algorithms, the models were trained on 75% of the data (500 verbatims) and were tested on the remaining verbatims (138 verbatims). The classification model was tested with a set of the original verbatims and the resulting accuracy was 85%.

To introduce a form of feedback and scoring inside the game, simulated social feedback was added showing number of "likes" and "friends" depending on the player responses. When players' maturity level increased, players gained "likes", and when they made positive evaluations of the opinions of NPC with a higher level of maturity than their own, they also gained friends.

Experimentation

Nine dilemmas were transposed in the Unity 3D environment. Three other dilemmas were used as pretest and three more as post-tests using the original SoMoral format (computerized). The setup for those tests were similar to the ones previously developed [1]. The dilemmas chosen to be part of the game and those chosen for the pretest and posttest were selected to be representative of different types and levels of difficulties, but it was also important that they would be sufficiently different from one another.

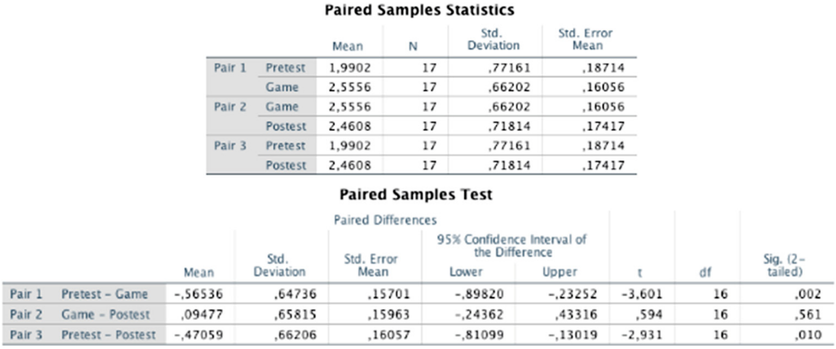
The aim of the study was two-fold: to measure the potential of the game for SMR maturity assessment and learning, and to assess its usability in terms of playability (the degree to which the game is fun to play and usable). The game was tested with 17 subjects (11 girls and 7 boys, 8–19 years).

Results

The first objective of the research was to measure the potential of the game to support users in developing a higher level of SMR maturity. A pretest and a posttest were used to compare the levels of maturity before and after playing the game. The results show that the mean results for the post test was significantly higher than the pre-test ($p = .01$) (Table 1). Table 1 also shows that there was a difference between pretest and the game, but no difference between the game and post-test. In fact, the scores in the post-test were slightly lower. This may be due to the post test being less socially immersive.

The second objective was to measure the usability of the game. The measure was based on the post-test questionnaire on user attitudes toward dimensions of immersion, playability (wanting to play again and telling friends) and learning something during the game. For those measures, no difference was found on the effect of sex. The amount of experience subjects had with social technologies and games did not appear to be

Table 1. Difference in mean maturity between pretest, game and post-test



Paired Samples Statistics

		Mean	N	Std. Deviation	Std. Error Mean
Pair 1	Pretest	1,9902	17	,77161	,18714
	Game	2,5556	17	,66202	,16056
Pair 2	Game	2,5556	17	,66202	,16056
	Postest	2,4608	17	,71814	,17417
Pair 3	Pretest	1,9902	17	,77161	,18714
	Postest	2,4608	17	,71814	,17417

Paired Samples Test

		Paired Differences							
					95% Confidence Interval of the Difference				
		Mean	Std. Deviation	Std. Error Mean	Lower	Upper	t	df	Sig. (2-tailed)
Pair 1	Pretest - Game	-,56536	,64736	,15701	-,89820	-,23252	-3,601	16	,002
Pair 2	Game - Postest	,09477	,65815	,15963	-,24362	,43316	,594	16	,561
Pair 3	Pretest - Postest	-,47059	,66206	,16057	-,81099	-,13019	-2,931	16	,010

related to their assessment of immersion, learning and playability in the game, nor was it related to learning in the game, as measured by the difference between pretest and post-test mean maturity level.

Discussion and Conclusion

We developed a preliminary prototype of a serious game for SMR. The assessment of the game suggests that it was appreciated by the players in terms of immersion, playability and impression of having learned something. Results also show that the game may encourages the development of higher levels of SMR maturity from pretest compared to the game, but also from pretest compared to post test. Results during the post-test appear to be lower than during the game, which might be related to the higher level of perceived immersion and also to the social simulation associated with the NPC and their opinions or the social feedback interface with number of “Likes”, “Dislikes” and “Friends”. Future work includes adding non-verbal feedback from NPC to make the game more immersive and responsive depending on players’ decisions and their evaluation of others.

References

1. Beauchamp, M., Dooley, J.J., Anderson, V.: A preliminary investigation of moral reasoning and empathy after traumatic brain injury in adolescents. Brain Inj. **27**(7–8), 896–902 (2013)
2. Chiasson, V., et al.: Assessing social cognition: age-related changes in moral reasoning in childhood and adolescence. Clin. Neuropsychol. **31**(3), 515–530 (2017)
3. Tato, A.: Convolutional neural network for automatic detection of sociomoral reasoning level. In: The 10th International Conference in Educational Data Mining, Wuhan, China (2017)

iMoodle: An Intelligent Moodle Based on Learning Analytics

Ahmed Tlili[1(✉)], Fathi Essalmi[1], Mohamed Jemni[1],
Maiga Chang[2], and Kinshuk[3]

[1] Research Laboratory of Technologies of Information and Communication & Electrical Engineering (LaTICE), Tunis Higher School of Engineering (ENSIT), University of Tunis, Tunis, Tunisia
ahmed.tlili23@yahoo.com, fathi.essalmi@isg.rnu.tn, mohamed.jemni@fst.rnu.tn

[2] School of Computing and Information Systems, Athabasca University, Athabasca, Canada
maiga.chang@gmail.com

[3] University of North Texas, 3940 N. Elm Street, G 150, Denton, TX 76207, USA
kinshuk@ieee.org

Abstract. Online learning is gaining an increasing attention by researchers and educators, since it makes students learn without being limited in time or space like traditional classrooms. However, this type of learning faces several challenges include the difficulties for teachers to control the learning process and keep track of their students' learning progress. Therefore, this paper presents an ongoing project which is an intelligent Moodle (iMoodle) that uses learning analytics to provide dashboard for teachers to control the learning process and make decisions. It also aims to increase the students' success rate with an early warning system for identifying at-risk students as well as providing real time interventions of supportive learning content as notifications.

Keywords: Learning analytics · Moodle · Online learning
Intelligent tutoring systems · At-risk students

1 Introduction

Distance educational systems have gained an increasing use within institutions in the 21st century since they offer e-learning options to students and improve the quality of traditional courses in classrooms. These e-learning systems, such as Modular Object-Oriented Dynamic Learning Environment (Moodle), provide students different types of activities, such as preparation of assignments and engagement in discussions using chats and forums. Moodle is one of the most well-known open-source e-learning systems which allows the development of interactive online courses [1].

However, the distributed nature of distance learning has raised new challenges. For instance, unlike classrooms, it becomes much harder for teachers in distance learning to supervise, control and adjust the learning process [2]. In massive open online courses,

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 476–479, 2018.
https://doi.org/10.1007/978-3-319-91464-0

where thousands of students are learning, it is very difficult for a teacher to consider individual capabilities and preferences. In addition, the assessment of course outcomes in Learning Management Systems (LMSs) is a challenging and demanding task for both accreditation and faculty [1]. Anohina [3] stated that it is necessary to provide a system intelligent and adaptive abilities so it could effectively take the teacher role. Researchers suggested using Learning Analytics (LA) to present important information about students online for teachers [2].

LA is often integrated into online learning environments, including Moodle, through the use of plugins. However, plugins usually require a considerable effort, most often involving programming, to adapt or deploy them [2]. This can limit their use by teachers. In addition, to the best of our knowledge, no plugin is reported online which provides real-time interventions for students for a better learning process. Therefore, this paper presents, in the next section, iMoodle – an intelligent Moodle based on a newly developed online LA system named Supervise Me in Moodle (SMiM), which: (1) provides dashboards for teachers to easily help them supervise their students online; (2) predicts at-risk students who may fail to pass their final exams; and, (3) provides real time interventions, as notifications, by providing supportive learning content for students while learning.

2 Framework for Intelligent Moodle (IMoodle)

Figure 1 presents the framework of the implemented iMoodle. During the learning process, the students' traces are collected in an online database and automatically analyzed in order to extract knowledge and provide real time interventions. A learning analytic system SMiM is developed using web technologies and integrated into Moodle as a Moodle block where teachers can easily access it and keep track of their students in each enrolled course. SMiM has three layers as follows:

(1) Privacy layer keeps students' traces safe with the login and password authentication method. In this context, to access the reports and information provided by SMiM, the teacher should have his/her session already active on iMoodle (i.e., the teacher has already entered his/her credentials to access iMoodle and chosen his/her courses). If not, the teacher will be redirected to the authentication interface.

(2) Analysis layer uses both data mining and visualization techniques to extract useful information for teachers. SMiM uses association rules mining based on Apriori algorithm, to identify early in the semester at-risk students within iMoodle who would likely fail their final exams of a particular course, hence increase academic success by providing early support.

(3) Reporting layer provides reports and real time interventions for the identified at-risk students while learning. SMiM provides dashboards for teachers to aid them control the learning process online and keep track of their students. In addition, if students failed to correctly finish a particular learning activity, iMoodle provides real time interventions, as notifications, by providing additional learning content support for students to further enhance their knowledge.

Furthermore, through the use of predictive modeling techniques, it is possible to forecast students' success in a course and identify those that are at-risk. Therefore, iMoodle, based on SMiM system, uses a predictive model (discussed in the analysis layer) as an early warning system for identifying at-risk students in a course and inform the teacher.

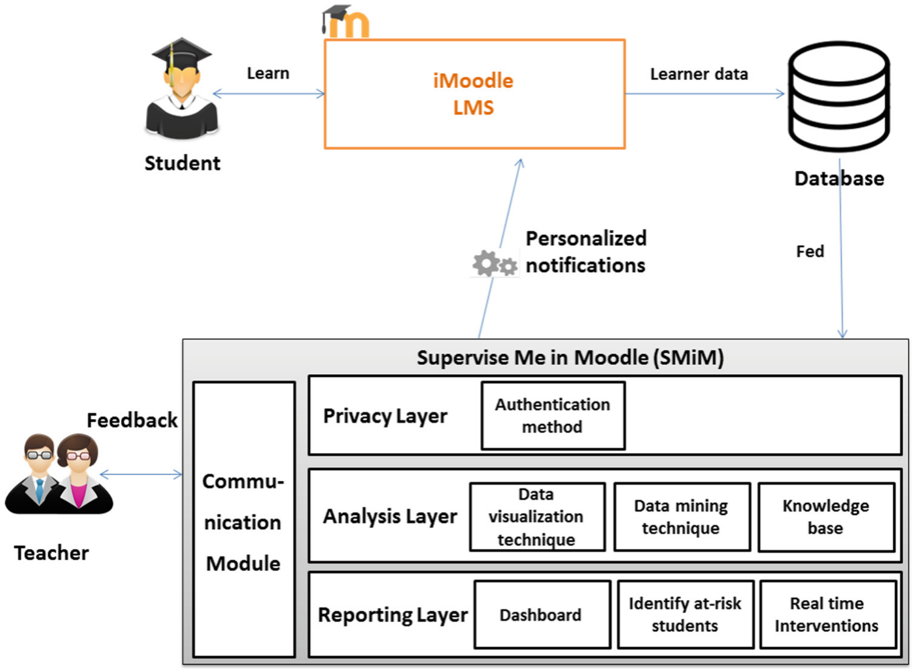


Fig. 1. The developed iMoodle Framework.

3 Conclusion

This paper presented a new intelligent version of Moodle (iMoodle) which aims to help teachers control the learning process online and keep track of their students. Future work could focus investigating the efficiency of iMoodle using the intervention layer in reducing the number of at-risk students and increasing academic success, in comparison with a classic Moodle.

References

1. Yassine, S., Kadry, S., Sicilia, M.A.: A framework for learning analytics in moodle for assessing course outcomes. In: Global Engineering Education Conference, pp. 261–266 (2016)
2. Vozniuk, A., Govaerts, S., Gillet, D.: Towards portable learning analytics dashboards. In: 13th International Conference on Advanced Learning Technologies, pp. 412–416 (2013)
3. Anohina, A.: Advances in intelligent tutoring systems: problem-solving modes and model of hints. J. Comput. Commun. Control **2**(1), 48–55 (2007)

Doctoral Consortium

Analysis and Optimization of Brain Behavior in a Virtual Reality Environment

Hamdi Ben Abdessalem(✉)

Département D'Informatique et de Recherche Opérationnelle,
Université de Montréal, Montréal H3C 3J7, Canada
benabdeh@iro.umontreal.ca

Abstract. The causes of humans' emotions change are multiple. In order to analyze them, we propose to follow the emotions of an individual in real-time during his interaction with a virtual environment. Then, we propose to intervene on the virtual environment through a neural agent in order to modify and improve the humans' emotional state. Finally, we propose a personal agent, which aims to personalize the environment in order to optimize humans' emotions.

Keywords: Intelligent agent · Virtual reality · Neurofeedback · EEG
Emotional intelligence

1 Introduction

The performance of users when interacting with learning systems or other types of programs varies according to their emotional states. Physiological measures of brain activity (EEG) [1] and eye tracking [2] provide better understanding of individual's emotions. Virtual reality helps the user immerse in the environment as if he was in a real one and that way his learning ability and performance will increase [3].

Changes in the virtual environment will cause a change in his emotional state and each modification can have a different impact on the emotional state. The negative emotional states of the user affect his cognitive state, for that, the modification of the emotional states in order to improve them will improve his cognitive state and thus his performance. Therefore, we need to detect the impact of the changes on the user's emotional states. However, sometimes the modification of the virtual environment are not enough to, modify the emotional state of the user. Thus, we need to learn from the link between changes on the virtual environment and changes in emotional states.

Therefore, we have three objectives: (1) Track in real time the emotional states of the user while interacting with the virtual environment in order to analyze his emotional states. (2) Modify the user's emotional states indirectly through the modification of the virtual environment in order to improve the user's emotional state and optimise his performances. (3) Observe the user's emotional reactions after each modification on the virtual environment in order to predict their impact on user's emotional states and thus, personalise the virtual reality environment to each user.

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 483–485, 2018.
https://doi.org/10.1007/978-3-319-91464-0

2 Methodology

In order to achieve our goals, we propose to create a neurofeedback system containing three components: a “Measuring Module” which responds to our first objective, a “Neural Agent” which responds to our second objective and a “Personal Agent” which responds to our third objective. Figure 1 illustrates the architecture of our neurofeedback system.

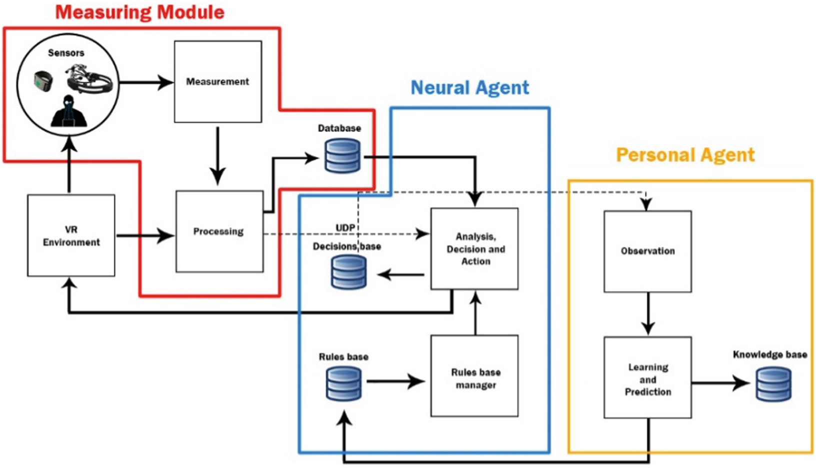


Fig. 1. Architecture of the neurofeedback system

The measurement module receives signals from sensors (EEG, eye tracking, etc.), analyzes them, and extracts the indices of emotional states. Then, this module sends these emotional states in addition to information about the virtual reality environment to the neural agent and stores them in a database for offline analysis.

The neural agent is an intelligent agent that receives the user’s emotional states from the measurement module and the information of the virtual environment, then it consults the rules base, which contains intervention rules, to intervene on the virtual environment and modify the emotional state of the user.

The personal agent is a cognitive agent that aims to adapt the virtual environment to the user. It observes the interactions between the users’ emotional states and the interventions on the virtual environment. Indeed, the personal agent observes the neural agent’s, learns from its interactions with the virtual environment and their impact on the emotional state of the user in order to create new intervention rules and adapt better the environment to the user. This agent runs in parallel with the neural agent to perform the learning and prediction tasks. The heavy learning computing performed by this agent does not affect the real-time execution of the neural agent and the entire neurofeedback system because it does not intervene directly on the virtual environment. The personal agent personalizes the virtual environment by modifying the neural agent’s rules base, which will then modify the environment.

3 Preliminary Results

We started by creating the measuring module and for that, we created the measurement component and the processing component in the module. After that, we integrated the Emotiv SDK EEG headset. In order to test this measuring module, we created a physics virtual reality game called "Inertia" which aims to improve the player's intuitive reasoning. We conducted experiments, involving 20 participants. We used frustration and engagement provided by the measuring module in order to assist the players [4]. Results showed that players' performance increased when adding assistance strategies.

Then, we created the neural agent and we created "AmbuRun" an adaptable virtual reality game in order to test this agent. We conducted experiments, involving 20 participants, in which the neural agent changes the speed of the game in order to affect excitement and changes the difficulty of the game which affects frustration. Results showed that when the agent adapts the game for the participant by changing speed and difficulty according to his excitement and frustration, it affects the level of his excitement and frustration in the right way [5].

Further work will aim to analyze the effect of each intervention with machine learning techniques to provide the personal agent with deeper adapting capabilities.

Acknowledgment. We acknowledge NSERC-CRD and Beam Me UP for funding this work.

References

1. Chaouachi, M., Frasson, C.: Mental workload, engagement and emotions: an exploratory study for intelligent tutoring systems. In: Cerri, S.A., Clancey, W.J., Papadourakis, G., Panourgia, K. (eds.) ITS 2012. LNCS, vol. 7315, pp. 65–71. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-30950-2_9
2. Ben Khedher, A., Frasson, C.: Predicting user learning performance from eye movements during interaction with a serious game. In: EdMedia: World Conference on Educational Media and Technology. Association for the Advancement of Computing in Education (AACE), pp. 1504–1511 (2016)
3. Biocca, F.: The cyborg's dilemma: progressive embodiment in virtual environments. J. Comput.-Mediat. Commun. **3** (2006)
4. Ghali, R., Ben Abdessalem, H., Frasson, C.: Improving intuitive reasoning through assistance strategies in a virtual reality game. In: The Thirtieth International Florida Artificial Intelligence Research Society Conference. AAAI, Florida, USA (2017)
5. Ben Abdessalem, H., Frasson, C.: Real-time brain assessment for adaptive virtual reality game : a neurofeedback approach. In: Frasson, C., Kostopoulos, G. (eds.) BFAL 2017. LNCS, vol. 10512, pp. 133–143. Springer, Cham (2017)

Enhancing EFL Students' Collaboration in a Blended Learning Environment: A Design-Based Research

Zexuan Chen[1,2](✉) and Jianli Jiao[2]

[1] School of Information Technology in Education,
South China Normal University, No. 55, West of Zhongshan Avenue,
Guangzhou, Guangdong Province, China
SerlinaChen@163.com
[2] School of Foreign Studies,
Southern Medical University, No. 1838, North of Guangzhou Avenue,
Guangzhou, Guangdong Province, China

Abstract. Collaboration is widely accepted as one of the essential skills for the 21st century. Based on social constructivist theory, collaboration has been gradually extended to synchronous and asynchronous online collaboration supported with Web 2.0 technologies. The purpose of this paper is to explore effective strategies to enhance EFL students' synchronous and asynchronous collaboration both distributed and face-to-face learning environments. Dillenbourg's (1999) four criteria (situation, interactions, processes and effects) for collaborative learning would be applied to develop proposed strategies that define the collaboration situation, encourage interactions during the collaboration, facilitate the collaboration processes and measure specific learning outcomes. Pérez-Sanagustín et al.'s (2012) 4SPPIces model would be used as a reference to further modify the proposed collaboration strategies in the blended learning environment. A design-based research would be conducted to test out the effects of the proposed strategies and shed light on rewarding modifications to the strategies. Participants would be 120 freshmen who are enrolled in a college English course in one of the universities in mainland China. A three-cycle of iterative experiment would be conducted to collect both qualitative and quantitative data. Among which, qualitative data include records of students' interactions and collaboration processes in class and out of class, student products, student feedbacks in the interviews; whereas, quantitative data would be composed of students' perceptions of the collaboration situation, effects on their collaboration capabilities.

Keywords: Collaboration · Blended learning
Design-based research · EFL students

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 486–489, 2018.
https://doi.org/10.1007/978-3-319-91464-0

1 Introduction

Collaboration is widely accepted as one of the essential skills for the 21st century (Morel 2014), as collaboration can have powerful effects on student learning (Lai, 2011), such as promoting academic achievement, personal development and student satisfaction etc. (Barkley et al. 2005).

Thanks to the rapid development of the Internet and Web 2.0 technologies, collaborative learning can so far occur both in and beyond the classroom, making collaboration in the blended learning environment possible (Pérez-Sanagustín et al. 2012). However, researches of this kind are albeit limited up till now, classroom teachers, especially EFL teachers, are still in lack of effective strategies to enhance collaboration in/out of class.

Therefore, the present paper would attempt to explore effective strategies to enhance EFL students' synchronous and asynchronous collaboration in a blended learning course, which consists of both distributed and face-to-face learning environments.

2 Literature Review

2.1 Collaborative Learning

Based on social constructivist theory, collaborative learning is a way in which individuals work closely together towards a common goal, adopting expertise and experiences and emphasizing co-creation and contributions from each member of the group (Gokhale 1995).

Aiming at clarifying the concept of "collaboration", Dillenbourg (1999) has put forward four criteria (situation, interactions, processes and effects) for collaborative learning. These criteria play an important role in informing learning designers and classroom teachers to develop strategies that clearly define the collaboration situation, successfully encourage interactions during the collaboration, facilitate the collaboration processes and effectively measure specific learning outcomes.

2.2 Collaboration in the Blended Learning Environment

"Blended Learning" is learning that "combine face-to-face instruction with computer mediated instruction", whereas blended learning environment is composed of distributed learning environments and face-to-face learning environments (Bonk & Graham, 2006: 5).

Thanks to the rapid development of communication technologies, collaboration has been gradually extended to include both synchronous and asynchronous collaboration in/out of class.

Some scholars argue that collaborative blended learning activities could be characterized with different representations due to different time (synchronous or asynchronous) and space (distributed or face-to-face) (Avouris et al. 2008; Pérez-Sanagustín et al. 2012; Siampou et al., 2014; etc.). Among them, Pérez-Sanagustín et al. (2012) has put forward a 4SPPIces model to guide educators in the design of Computer-supported Collaborative Blended Learning (CSCBL). They categorize the Space Factor (S) as virtual space and physical space. According to them, collaboration in the virtual space are mostly distributed collaboration, while the physical space would support both non-Electronic and Electronic collaboration. They point out that the virtual and physical spaces could be connected via electronic components, i.e. synchronous or asynchronous in either distributed or face-to-face learning environments could be made possible and could be facilitated with the help of Web 2.0 technologies.

Previous studies have been conducted on distributed asynchronous collaboration (Bates, 2015), distributed synchronous collaboration (Higley, 2013; Siampou et al., 2014; etc.), distributed asynchronous collaboration in the blended learning environments (Chen & Hou, 2014), collaboration in the blended learning environments (Capponi et al. 2010; Pérez-Sanagustín et al. 2012). However, few studies have been conducted to research strategies on enhancing both synchronous and asynchronous collaboration in distributed or face-to-face learning environments. Therefore, the present paper would.

3 Methodology

The present paper would conduct a design-based research to develop a set of implementing strategies to enhance EFL students' collaboration in a blended learning environment, then test out the effects of the proposed strategies and shed light on rewarding modifications to the strategies.

3.1 Research Question

Research question of the present paper is: How to enhance EFL students' collaboration in a blended learning environment? To make the research question more answerable, it is subdivided into four specific questions as follows,

(1) How to define the collaboration situation?
(2) How to encourage interactions during the collaboration?
(3) How to facilitate the collaboration processes?
(4) How to measure specific learning outcomes?

3.2 Participants

Participants would be 120 freshmen who are enrolled in a college English course in one of the universities in mainland China.

3.3 Data Collection

A three-cycle of iterative experiment would be conducted to collect both qualitative and quantitative data. Among which, qualitative data include records of students' interactions and collaboration processes in class and out of class, student products, student feedbacks in the interviews; whereas, quantitative data would be composed of students' perceptions of the collaboration situation, effects on their collaboration capabilities.

References

Dillenbourg, P.: What do you mean by collaborative learning? In: Dillenbourg, P. (ed.) Collaborative-Learning: Cognitive and Computational Approaches, pp. 1–19. Elsevier, Oxford (1999)

Pérez-Sanagustín, M., Santos, P., Hernández-Leo, D., Blat, J.: 4SPPIces: A case study of factors in a scripted collaborative-learning blended course across spatial locations. Comput.-Support. Collab. Learn. **7**, 443–465 (2012)

Leveraging Mutual Theory of Mind for More Human-Like Intelligent Tutoring Systems

Bobbie Lynn Eicher(✉), David Joyner, and Ashok Goel

Design & Intelligence Laboratory, School of Interactive Computing,
Georgia Institute of Technology, Atlanta, GA 30332, USA
{beicher3, david.joyner, ashok.goel}@gatech.edu

Abstract. Educational interactions are a fundamentally collaborative act between the student and the theory, where each is attempting to understand and interpret the behavior of the other. Intelligent tutoring systems can be made more effective by designing them with collaboration as a key consideration, creating systems that don't just build a model of the student but also attempt to improve the student's self-understanding and understanding of the tutoring system itself.

Keywords: Intelligent tutoring systems · Theory of mind
Cognitive science · Online education

Background

Theory of Mind

The concept of theory of mind arose out of research on chimpanzees, in which Premack and Woodruff noted that they had the ability to understand that it is possible for different individuals to have differing understandings and beliefs about the world [6]. This has since been studied mainly in the context of human children and adults with atypical cognition, such as those on the Autism spectrum [1].

In studies among neurotypical adults, the measured level of skill individuals have at theory of mind is an indicator of how well they will perform when asked to collaborate on tasks [7, 8]. This result held up in further experiments, even in settings where the groups doing the collaborating were operating entirely in a virtual setting and could not see one another or rely on body language and facial expressions as cues [3]. This is interesting from an educational perspective due to the critical role that the ability to read one another and make collaborative decisions plays in effective teaching and learning.

AI Teachers in Online Classes

Growth in online learning has increased the pressure to find ways to leverage technology to provide students with accurate and appropriate answers to their questions at any time of the day or night. In online courses offered through Georgia Tech's College of Computing, course teams found themselves working in online class forums where

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 490–492, 2018.
https://doi.org/10.1007/978-3-319-91464-0

the number of interactions that had to be read and handled was six times as large as those in the traditional campus offerings and continuing to grow. This led to the development of Jill Watson as an agent capable of monitoring the same forums and threads used by the human teaching team, reading the questions that the students posed, and providing answers with roughly the same frequency, accuracy, and authenticity as many of the members of the human teaching team; many students had no idea that Jill Watson was not a human until it was announced at the end of the semester [4].

Bringing Theory of Mind to Tutoring Systems

This work is focused on analyzing the ways that students and teachers leverage theory of mind to improve the value of their interactions, and how this skill may allow virtual teaching assistants and other intelligent tutors to aid that is more valuable and appropriate for the needs of students. In fact, it should be regarded as a mutual process where each side of the interaction is both attempting to build an understanding of the other and attempting to monitor the other side's beliefs to offer attentional corrections and improvements to understanding.

This approach originated as a part of a project focused on building a set of models representing the different ways in which students have been observed to incorrectly understand the way a compiler or interpreter implements the behavior of assignment statements [2, 6]. After building the models, we went on to give the tool the ability to represent the incorrect models alongside the correct one at each step of execution for small snippets of code, and also to attempt to predict what kind of misunderstandings a student might have based on their stated expected output so that the system could provide corrections that specifically target their own mistaken believes about the workings of the computer itself [2].

Therefore, the goal is to enable enhanced collaboration by not just guessing at what a human really meant or attempting to correct them, but actively seeking to aid them in identifying the specific point at where their expectations went wrong and how they did so, in the interest of making improvements.

Ongoing and Future Work

We are currently working to build this idea into our approach for virtual teaching assistants for both graduate and undergraduate courses offered online, and into tutoring tools designed to assess students on very specific topics with questions that accept open-ended input. The open-ended nature of the answer space is both an opportunity and a challenge, because it allows students to provide information on what they believed to be correct, rather than a best guess based on a limited pool of possibilities (as in multiple choice). Richer and more specific information comes at the cost of greater complexity in processing the response and selecting a reaction.

To further improve on our ability to determine where student misunderstandings occur and what their nature is, we're working on tools to compile the data from a variety of existing exercises that are in a more standard format as well. While these are individually less rich as a form of input, as a group they represent a large pool of existing data that we can use to determine where it makes sense to invest time in creating improved exercises and serve as a useful start on identifying likely errors requiring corresponding responses and coaching.

With this approach, we hope to be able to leverage the size of online classes to improve our approach. Gathering adequate information on how students understand and misunderstand each topic within a course is the biggest challenge that we've faced so far. We believe that the scale of online courses (both for-credit and MOOCs) will allow us to leverage small tutors that accept open-ended input to gain deeper insight into exactly how and where students are misunderstanding material and how best to provide corrections, guide the students through a course, and help the students to better understand how they can take advantage of the set of tools available as part of a course to improve their own educational experiences.

References

1. Baron-Cohen, S., Wheelwright, S., Hill, J., Raste, Y., Plumb, I.: The reading the mind in the eyes test revised version: a study with normal adults, and adults with asperger syndrome or high-functioning autism. J. Child Psychol. Psychiatry **42**(2), 241–251 (2001)
2. Eicher, B., Cunningham, K., Marissa Gonzales, S.P., Goel, A.: Toward mutual theory of mind as a foundation for co-creation. In: Presented to the International Conference on Computational Creativity, Co-Creation Workshop, June 2017
3. Engel, D., Woolley, A.W., Jing, L.X., Chabris, C.F., Malone, T.W.: Reading the mind in the eyes or reading between the lines? Theory of mind predicts collective intelligence equally well online and face-to-face. PLoS ONE **9**(12), 1–16 (2014)
4. Goel, A., Polepeddi, L.: Jill Watson: a virtual teaching assistant for online education. In: Presented to the Learning Engineering for Online Learning Workshop, Harvard University, June 2017. (To appear as a chapter in Dede, C., Richards, J., Saxberg, B., (eds.) (in preparation) Education at Scale: Engineering Online Teaching and Learning. Routledge, NewYork (2017))
5. Goel, A., Joyner, D.: An experiment in teaching artificial intelligence online. J. Scholarsh. Technol.-Enhanc. Learn. **1**(1) (2016)
6. Ma, L.: Investigating and improving novice programmers' mental models of programming concepts. Ph.D. Dissertation, University of Strathclyde (2007)
7. Premack, D., Woodruff, G.: Does the chimpanzee have a theory of mind? Behav. Brain Sci. 4 (4), 515–629 (1978)
8. Sorva, J.: Notional machines and introductory programming education. Trans. Comput. Educ. **13**(2), 8:1–8:31 (2013)
9. Woolley, A.W., Chabris, C.F., Pentland, A., Hashmi, N., Malone, T.W.: Evidence for a collective intelligence factor in the performance of human groups. Science **330**(6004), 686–688 (2010)

Effect of Learning Support System for Feature Words on Group Learning

Shun Okuhara[1,2(✉)] and Takayuki Ito[2]

[1] Fujita Health University, Toyoake 470-1192, Japan
sokuhara@fujita-hu.ac.jp
[2] Nagoya Institute of Technology, Nagoya 466-8555, Japan

Abstract. This paper proposes a system to support learning from Okapi BM25 feature words. When a learner does not understand difficult feature words, he/she cannot reach consensus in discussions. Automatic generation of feature words for teaching could greatly help teachers create instruction for an intelligent tutoring system for group learning. When students have inadequate knowledge, teachers need to intervene to teach them. However, a teacher cannot intervene with large groups. Therefore, the researchers/we prototyped a system to support learners rather than teachers in group learning. We experiment to analyze the effect of the prototype system for creating feature words. This experiment's compared learners who used the system (group A) with those who did not (group B). The learners discussed the job placement problem. We found that the Group A had a higher score than Group B. There was a significant difference between groups. Results show that the system correctly determines feature words and learning effects on students were confirmed in group learning.

Keywords: Group learning · AutoTutor · Okapi BM25

1 Introduction

This study developed a learning-support system as a substitute for an instructor, which explains important words in a discussion among learners. Especially in group learning, this developed system provides learning support by explaining important words that frequently appear in discussions (hereafter, feature words). This study provides learning support, using the concept of AutoTutor [1], an intelligent tutoring system that holds conversations with humans in natural language. Learning based on AutoTutor follows an interactive format where learners respond to questions. This study conducts learning support, based on the concept of AutoTutor, by explaining feature words to learners. In learning support with the existing AutoTutor, topics such as subjects would be pre-registered by an instructor as data for dialogue and then provided to learners. It is difficult to create data in advance for learning whose topic changes at any time, like discussions, and because of that, it is difficult to incorporate AutoTutor into the real classroom. Hence, this study proposes a mechanism that can explain feature words independent of topics discussed among learners. This proposal introduces a pedagogical agent as a mechanism where an instructor does not predetermine the words to be explained as the existing AutoTutor does. This study implemented a pedagogical

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 493–495, 2018.
https://doi.org/10.1007/978-3-319-91464-0

agent to intervene in learners, based on Okapi BM25 [2], which is widely used for information retrieval and document recommendation. Okapi BM25 is an index to identify how characteristic the specific words that appear in a document are. The objective of this study is to develop an agent that can flexibly explain feature words to participants while the content of a discussion changes constantly. Therefore, this study will evaluate, using Okapi BM25, the pedagogical agent that can intervene even if an instructor does not enter the content of learning into the system in advance and verify its utility.

2 Design of Studies

In this study, we conduct an intervention experiment using two groups. The first group is one with an intervention, which, when a word deemed to be a feature word by its features is uttered, explains it using a teaching agent. The second group is a group without intervention, which conducts discussions in an environment that enables students to look up words unknown to them on the internet. For the feature words, we use words selected from the dialogue data of a discussion among 74 people who participated and discussed under the same theme as in this experiment. The experiment was conducted with 32 people in the group A with an intervention and 32 in the group B without an intervention. The groups randomly select 3–5 students who participate in the discussion. The issues discussed in this experiment are related to the placement for job-opening information at Hello Work (hereafter, the job placement service problem). The challenge is to select job-opening deemed to be the most suitable for the job seekers as Hello Work staff. In the placement task for job-opening, students are distributed the job opening information, answer sheets and prefectural maps of three job seekers looking for the position as medical clerks, in which the desired working conditions (hereafter, job seekers' conditions) of each of them are mentioned. First, from the sentences indicating the job seekers' conditions, we give them the task of entering each job seeker's information in a table in an easily identifiable manner. The next task is for students to enter in the answer sheet the name of the company whose conditions are most suitable for three job seekers out of the six job-opening information, and mention at the end the reason why the placement was recommended. In order to measure the learning effect of the system prototyped from the above tasks, this research conducted tests before the above task to measure the degree of understanding of words related to job-opening information (hereafter, pre-tests), and tests to measure the degree of comprehension of the similar contents after the experiment (hereafter, post-tests).

3 Results

The experimental results describe the values of change in scores on the pre-test and the post-test taken by the group with intervention and the group without intervention (hereafter, the values of change in test scores). In this study, the group A with intervention and the group B without intervention were investigated based on a t-test for the values of change in test scores. The results of the values of change in scores are shown in Fig. 1. The average scores on the pre-test were 10.09 in the group A with intervention and 10.06 in the group B without intervention. Then, the average scores on the post-test were 13.03 in the group A with intervention and 11.90 in the group B without intervention. The values of change in test scores were 2.93 in the group A with intervention and 1.84 in the group B without intervention, and it was found that the value in the group A with intervention was high. Furthermore, as a result of a t-test for the values of change in test scores in the groups with and without intervention, the p-value was 0.046, and there was a significant difference between the groups with and without intervention.

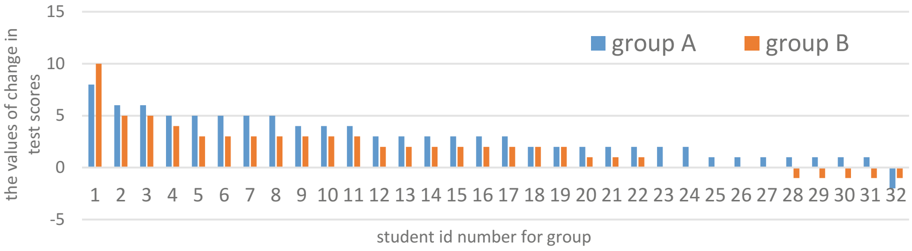


Fig. 1. The values of change in test scores.

4 Conclusion

The objective of this study is to develop a system as substitute for an instructor, which can provide discussion support by flexibly explaining important words in a discussion whose content changes constantly. Therefore, this study developed a system where a pedagogical agent intervenes by using important words. This study confirmed that when a pedagogical agent made an explanation to learners using important words in actual group learning, the learners have learned the meanings of the words, and that this is effective. However, this study has not closely examined which timing of intervention has the most profound learning effect although explaining what learners do not know has shown some learning effect. Therefore, it is necessary to investigate which conditions of intervention could contribute to the growth of learners' knowledge, and this merits further research

Acknowledgements. I would like to take this opportunity to thank Koichi Muto for years of collaboration and advice. This work is supported by the Sasakawa Scientific Research Grant.

References

1. Büttcher, S., Clarke, L.A.C., Cormack, V.G.: Information Retrieval: Implementing and Evaluating Search Engines. The MIT Press (2010)
2. Graesser, A,C., VanLehn, K., Rose, P,C., Jordan, W,P., Harter, D.: Intelligent tutoring systems with conversational dialogue. AI Mag. **22**, 39–51 (2001)

A New Approach to Testing Children with Autism Spectrum Disorder Using Affect

Veronica Rivera(✉)

University of California Santa Cruz, Santa Cruz, CA 95060, USA
veariver@ucsc.edu

Abstract. In order to qualify for special education services, elementary school children with Autism Spectrum Disorder (ASD) are given a myriad of standardized tests. However, even when they have high cognitive abilities they often have difficulty answering test questions due to the nature of their disability. This causes them to underperform and not qualify for the services that would best suit their skills. The present proposal details a research plan to create an intelligent testing system that attempts to motivate the student upon detecting boredom and low levels of engagement.

Keywords: Affective computing · Intelligent tutoring system
Autism spectrum disorder · Learner motivation · Engagement
Standardized testing

1 Introduction and Problem Description

School-age children diagnosed with Autism Spectrum Disorder (ASD) are given several standardized tests in preschool and elementary school such as the Test of Visual Perceptual Skills, to measure their abilities in areas such as visual discrimination and sequential memory. Test results are a crucial factor used by school psychologists to recommend appropriate special education services. However, children with ASD often have difficulties taking standardized tests not because they lack the appropriate skills, but because of the nature of their disability. They exhibit deficits in joint attention, lack of motivation in answering test questions, and difficulty interacting with the examiner. These difficulties negatively impact their future academic progress by causing them to underperform on assessments, resulting in inappropriate educational placement.

Many of these problems arise from the format in which the tests are administered. Current tests are often given orally with verbal instructions, which are more difficult for children with ASD to understand than visual instructions [7]. This may cause low motivation to complete tests. Additionally, these tests are not engaging, lowering motivation even further.

However, most children with ASD show an affinity towards content displayed on computers [6]. Because of their eagerness to utilize technology, children with ASD

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 496–498, 2018.
https://doi.org/10.1007/978-3-319-91464-0

could greatly benefit from intelligent tutoring systems to enhance their education. Emotion and affect have been used to make intelligent tutoring systems more personal and engaging [1]. Such research is backed by studies demonstrating that emotions profoundly affect students' academic performance and ability to problem solve [5]. Most existing systems have been designed for and tested on neurotypical students. Because children with ASD exhibit increased emotional responses, we hypothesize that their performance on standardized tests would be further affected by their emotional state.

Intelligent testing systems exist for neurotypical individuals, such as the one in [3]. However, the proposed intelligent testing system offers a new approach by using automatic emotion detection to engage students during the test and targeting a group of users that is not yet well represented by existing systems.

2 Proposed Solution

An interactive testing method that responds to decreases in levels of engagement would enable children with ASD to perform to the best of their abilities on standardized tests. Two research questions driving this project are: (*1) What factors motivate a child with ASD to complete academic tasks? (2) How can boredom and changes in engagement level most accurately be detected?*

The test will be administered to students on a computer with a webcam, which will capture displayed emotions for the duration of a given test. It will use supervised learning methods such as those in [2] on the captured facial expressions to detect boredom and low levels of engagement. The goal of the proposed system is to be a tool for school psychologists and other educators to better serve children with ASD, not to replace the important role these professionals play. Therefore, the system will not make final placement decisions regarding appropriate educational services, but will provide test results and data about changes in a child's motivation.

Results from the emotion recognition software will be used to create an interactive testing system that responds to low levels of engagement and perceived boredom. It will attempt to motivate the child by adjusting the difficulty level of questions presented and using supportive comments and animations. A human cartoon animated with movement and sound effects will be present in a corner of the screen throughout the entire test, providing encouraging messages. It has been shown in past research that although children with ASD typically have poor face processing abilities, using animated cartoon figures in an intelligent tutoring system for learning vocabulary can provide benefits for many of them [4]. The authors of [4] hypothesized that these results can be extended to other educational settings. Additionally, instructions will be presented in a visual format.

The proposed system will be evaluated using a control group and an experimental group. The control group will be given a test such as the Test of Visual Perceptual Skills, and the experimental group will be given a test using the proposed intelligent testing system. The proposed system will be considered successful if students in the experimental group exhibit higher scores and generally enjoyed working with the system, as determined by a post-test survey. Confounding factors to take into account include a student's base intelligence, ability to use technology, and level of ASD.

3 Concluding Remarks

Although the proposed system is in its early stages, it holds great potential to open up new perspectives not just for improving the educational opportunities of children with ASD, but also for promoting inclusivity of individuals with special needs in existing intelligent tutoring systems. In the near future we will collaborate with special education teachers and school psychologists to create test content and revise the design plan before implementation of the software. We will also experiment with adding EEG sensors in the form of a wearable device to more accurately detect boredom.

References

1. Bosch, N., Chen, Y., D'Mello, S.: It's written on your face: detecting affective states from facial expressions while learning computer programming. In: Trausan-Matu, S., Boyer, K.E., Crosby, M., Panourgia, K. (eds.) ITS 2014. LNCS, vol. 8474, pp. 39–44. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-07221-0_5
2. D'Mellow, S., Jackson, T., Craig, S., Morgan, B., Chip-Man, P., White, H., Person, N., Kort, B., El Kaliouby, R., Picard, R., Graesser, A: AutoTutor detects and responds to learners affective and cognitive states. In: Workshop on Emotional and Cognitive Issues at the International Conference on Intelligent Tutoring Systems. Montreal, Canada (2008)
3. Kozierkiewicz-Hetmańska, A., Nguyen, N.T.: A computer adaptive testing method for intelligent tutoring systems. In: Setchi, R., Jordanov, I., Howlett, R.J., Jain, L.C. (eds.) KES 2010. LNCS (LNAI), vol. 6276, pp. 281–289. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15387-7_32
4. Massaro, D.W., Bosseler, A.: Read my lips: the importance of the face in a computer-animated tutor for vocabulary learning by children with autism. Autism **10**(5), 495–510 (2006). https://doi.org/10.1177/1362361306066599
5. Pekrun. R., Linnenbrink-Garcia, L.: Handbook of Research on Student Engagement. Springer, Boston (2012). https://doi.org/10.1007/978-1-4614-2018-7
6. Ploog, B.O., Scharf, A., Nelson, D., Brooks, P.J.: Use of computer-assisted technologies (CAT) to enhance social, communicative, and language development in children with autism spectrum disorders. J. Autism Dev. Disord. 301–322 (2013). https://doi.org/10.1007/s10803-012-1571-3
7. Quill, K.A.: Instructional considerations for young children with autism: the rationale for visually cued instruction. J. Autism Dev. Disord. **27**(6), 697–714 (1997)

Online Course Design to Support Learning Analytics in Facilitating Personalized Academic Help

Hongxin Yan[1(✉)] and Kinshuk[2(✉)]

[1] University of Eastern Finland, FI-80101 Joensuu, Finland
hongya@student.uef.fi
[2] University of North Texas, Denton, TX 76203, USA
kinshuk@ieee.org

Abstract. Online higher education is growing but carries some inherent difficulties for students. As many students do not actively seek help when stuck in learning, institutions should consider providing academic help in a proactive way. Since students are different and learn differently, personalized help is expected for the effectiveness. Leaning analytics (LA) is the technology that could be used to identify who are experiencing academic difficulties and what academic help is needed. Currently most LA systems are not functional to recommend detailed instructive feedback on how to improve a student's learning, and they also remain a challenge on data collecting. A research question is proposed from the course design perspective: how online courses can be designed in a way that supports learning analytics in facilitating personalized academic help. Some preliminary research has been conducted to answer this question.

Keywords: Course design · Academic difficulties · Personalized academic help LMS · Learning analytics · Dropout · Online higher education

1 Introduction

Online higher education has been rapidly growing and becoming one of the top industries in the world. While this education model provides students with certain learning flexibilities, it manifests some inherent difficulties for students [8]. For example, it requires higher self-regulated learning skills in students than traditional education. The trouble of understanding a concept or solving an academic problem is referred as academic difficulties in this study. If students are stick in learning or struggling with the academic difficulties but not getting help, it could lead to learning incompletion or dropout, a serious issue in online higher education. Hence, timely support from institutions is crucial to help those at-risk students [8].

Students are different and learn differently. Even provided with a well-designed course and having high motivation to learn, some students would still encounter the academic difficulties. Academic difficulties can be caused by different factors, such as the lack of prior knowledge, ineffective pedagogy design, insufficient practice, etc.

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 499–502, 2018.
https://doi.org/10.1007/978-3-319-91464-0

Therefore, even students are stuck at a same learning point, they might need different academic help, such as additional resources, more explanation, etc.

The distance in online education creates a barrier for instructors and students to know each other. Most institutions are relying on students' help seeking to provide the academic help. However, studies show many students are not actively seeking help when they need [6]. In some cases, the very students who need help the most seek it the least [1], or some avoid seeking academic help even after struggling fruitlessly on their own [5]. Therefore, institutions should consider providing academic help in a proactive way. Intelligent computing technologies, such as the learning analytics, could help.

The learner-produced data in online environments (e.g. click, page reading, social interaction, grade, learning products) provide valuable insight of how students learn [9]. "Learning Analytics is the measurement, collection, analysis and reporting of data about learners and their contexts, for purposes of understanding and optimizing learning and the environments in which it occurs" [9, p. 32]. As Chatti et al. [2] stated that based on student's current activities and performance, learning analytics can be used to provide proactive intervention for students who may need additional assistance. Given the potential of learning analytics in this regard, the goal of this study is to explore how to use learning analytics to facilitate personalized academic help in a proactive way.

2 The Research Gap and a Challenge

While the field of learning analytics is constantly developing new ways to analyze educational data and track student's performance [2], most research tends to focus on the trace data of interactions with learning management system (LMS) [3], such as access to learning materials, login frequency, online spent time. Predicting systems using such data source mainly focus on static prediction of a single academic outcome – dropout, failure or success [3], so they are not functional to identify what academic difficulties that students are experiencing and therefore not able to recommend detailed instructive feedback on how to improve students' learning [2].

Another big challenge in this field is where and how to get the necessitated learning data for analytics to operate. Student activities are often distributed over open and increasingly complex learning environments today, therefore, not only is data from a wide variety of sources distributed across space, time, and media, but also can it come in different formats [2]. How to aggregate and integrate raw data from multiple and heterogeneous sources is a big challenge. Also, data privacy is another concern with open data sets, hence, collecting such a volume of data from student's daily life could be regarded as an invasion of privacy [2].

Currently, LMS systems are widely used in academic institutions around the worldwide. LMSs capture and store large amounts of sophisticated user activity and interaction data in real time, which can be mined at any stage of course progression to find out learner behavior [4]. Those user activity and interaction data are generated out of the activities designed in LMSs, such as pre-tests, surveys, forums, formative assessment, etc. With those activities designed in LMSs, no matter where and how students learn, students will come back to the LMSs and interact with the activities.

Therefore, it would be ideal if the course is designed in a way that sufficient learning data can be generated in a collectable and analyzable format for learning analytics to identify who are experiencing academic difficulties and what academic help is needed.

3 Research Questions

Based on the research gap and the challenge for learning analytics to support personalized academic help, the research question of this study is described in the following:

For online courses in higher education, what learning activities can be designed in LMSs to support learning analytics in identifying which students are experiencing academic difficulties and what effective academic help is needed by individuals?

This research question is divided into the following sub-questions: (a) what learning data indicates a student is experiencing academic difficulties; (b) what learning data reveals the type of academic help needed by the student; (c) what learning activities can be designed in LMSs to generate those learning data in (a) and (b); (d) how learning analytics analyzes those learning data to identify who needs the academic help and what academic help is needed. In the meanwhile, the learning activities should be designed in a pedagogically sound way. As different disciplines have distinct pedagogies [7], this research will focus on the STEM disciplines and a physics course is in consideration.

So far, some preliminary research has been done: interviewed some experts and academics in online education about the importance of this research question; revised the research proposal several times with the advice of my supervisor; systematic literature review is being conducted on personalized learning, academic difficulties and learning analytics;

References

1. Aleven, V., Koedinger, K.R.: Limitations of student control: do students know when they need help? In: Gauthier, G., Frasson, C., VanLehn, K. (eds.) ITS 2000. LNCS, vol. 1839, pp. 292–303. Springer, Heidelberg (2000)
2. Chatti, M.A., Lukarov, V., Thüs, H., Muslim, A., Yousef, A.M.F., Wahid, U., Greven, C., Chakrabarti, A., et al.: Learning analytics: challenges and future research directions. eleed, (10). (urn:nbn:de:0009-5-40350) (2014)
3. Gašević, D., Dawson, S., Siemens, G.: Let's not forget: learning analytics are about learning. TechTrends **59**(1), 64–71 (2015)
4. Macfadyen, L.P., Dawson, S.: Mining LMS data to develop an "early warning system" for educators: a proof of concept. Comput. Educ. **54**(2), 588–599 (2010)
5. Karabenick, S.A., Newman, R.S. (eds.): Help Seeking in Academic Settings: Goals, Groups, and Contexts. Routledge (2013)
6. Kinshuk: Designing Adaptive and Personalized Learning Environments. Interdisciplinary Approaches to Educational Technology. Routledge (2016). ISBN-10: 1138013064
7. Lindblom-Ylänne, S., Trigwell, K., Nevgi, A., Ashwin, P.: How approaches to teaching are affected by discipline and teaching context. Stud. High. Educ. (2006)

8. Paul, R.: If student services are so important, then why are we cutting them back? In: Sewart, D., Daniel, J.S. (eds.) Developing Distance Education (1988)
9. Siemens, G., Long, P.: Penetrating the fog: analytics in learning and education. EDUCAUSE Rev. **46**(5), 30 (2011)

Workshops

C&C@ITS2018: International Workshop on Context and Culture in Intelligent Tutoring Systems

Valéry Psyché[1](✉), Isabelle Savard[1](✉), Riichiro Mizoguchi[2,3], and Jacqueline Bourdeau[1](✉)

[1] LICEF Research Center, TELUQ University, Québec, Canada
cc-its2018@teluq.ca

[2] Research Centre for Service Science, Japan Advanced Institute of Science and Technology (JAIST), Nomi, Japan

[3] Laboratory for Applied Ontology (LOA), ISTC-CNR, Trento, Italy

With the internationalization of education, the need for adaptation and flexibility in ITS and other learning systems has never been more pressing, extending to many levels and fields including: the international mobility of learners, teachers and researchers; the integration of international, intercontextual and intercultural dimensions in instructional programs (from primary to higher education and continuing professional development), as well as in the designs, methods, techniques and tools that support them; the international mobility of education viewed through the lens of today's new reality of mass open online courses accessible by a diverse range of learners around the world facilitated by ubiquitous, mobile and cloud learning systems.

In this sense, there is a need for more research about context and culture in intelligent tutoring systems. Teachers and researchers need to develop new adaptation skills and embrace diverse contexts and cultures as well as leverage this diversity to foster the transfers that can enhance learning. Clearly therefore, it is important to make room for this diversity in curricula and learning systems and integrate transfer and adaptation concerns into pedagogical practice. But how can we do this concretely? How can we best manage this complexity and leverage this diversity? How can this materialize in the ITS field, and what are the benefits?

One of the main focuses of current research is to define the boundaries of context and culture (C&C) as a theoretical concept and what constitutes the best methods, techniques and tools in order to collect, analyze and model it from an adaptive learning perspective. Until recently, C&C modelling was considered an intrinsic part of the various classical ITS architecture models. Aspects of C&C were therefore partially covered under the domain, learner, pedagogical and communication models. Now, however, the advent of big data in education and significant innovations in artificial intelligence are opening new doors for us to analyse and model C&C differently, if we are able to take advantage of the information available through the learning analytics process. Big data offers an exciting opportunity for us to look at C&C modelling for ITS through a new lens. Do we need a fifth model? Should we view it as another layer in the ITS architecture? Let's start thinking about it. In today's era of adaptive learning delivering anything learners need, anywhere and at any time, the potential for context

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 505–506, 2018.
https://doi.org/10.1007/978-3-319-91464-0

and culture-aware ITS could be huge. What would knowledge representation and reasoning mechanisms look like in ITS? What kinds of limits might C&C represent for ITS? How can we identify or measure these limits? Can ocular and biometric measurement play an instrumental role? What are the logical next steps in terms of conducting studies about context and culture-aware ITS and gathering and analyzing data about context and culture?

Learning Analytics: Building Bridges Between the Education and the Computing Communities

Sébastien Béland[1], Michel Desmarais[2], and Nathalie Loye[1]

[1] Université de Montréal, Quebec, Canada
Sebastien.beland@umontreal.ca
[2] Polytechnique Montréal, Quebec, Canada

Description of the Workshop

The Learning Analytics (LA) and Educational Data Mining (EDM) fields have generated a wealth of research over the last decade, including two yearly conferences and two scientific journals. However, these topics are relatively new in the field of educational science.

This workshop aims to bring together researchers and practitioners to share their perspective on how this research has impacted the education field. Among the questions we wish to address is whether the two communities have a common perspective of the LA and EDM fields, and whether their expectations converge toward a common set of requirements. We also would like to address the perceived contributions of LA/EDM to the Educational community and to the Technology Enhanced Learning field, including MOOCs and the range of applications that foster means of self-driven learning.

Many topics are related to the idea of building bridges between the education and the computing communities. Here are some examples of interest:

- How teachers concerns can inspire further developments in LA and/or EDM?
- How to improve student's assessment using LA and/or EDM?
- How to improve educational management using LA and/or EDM?
- What are the biggest challenges of building bridges between the education and computing communities? What are the limits of the collaboration between these communities?
- Examples of collaborations between the education community and computing community.

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, p. 507, 2018.
https://doi.org/10.1007/978-3-319-91464-0

Exploring Opportunities for Caring Assessments

Diego Zapata-Rivera[1](✉) and Julita Vassileva[2]

[1] Educational Testing Service, Princeton, NJ 08541, USA
DZapata@ets.org
[2] University of Saskatchewan, Saskatoon, SK, Canada

1 Introduction

The notion of intelligent systems that "care" about students is at the center of ITS research [1]. A variety of adaptive learning systems that "care" have been developed in the past [2].

Caring assessment systems are defined as systems that provide students with a positive assessment experience while improving the quality of evidence collected about the student's knowledge, skills and abilities (KSAs) [3].

Taking a test is typically a stressful situation, and many people underperform due the stress. Caring assessment systems take into account assessment information from both traditional and non-traditional sources (e.g., student emotions, prior knowledge, and opportunities to learn) to create situations that students find engaging, and to collect valid and reliable evidence of students' KSAs.

Taking a test is not just a passive mechanism for assessing how much people know. It can actually help people learn, and it works better than a number of other studying techniques [4]. Caring formative assessment can be done by a computer system or by peer-learners. Developing systems or approaches (e.g. games) that support learners test each other in a constructive way, is a new and promising direction of research.

This workshop is a timely and relevant event for the ITS and assessment communities. New assessments for skills such as problem-solving, collaboration, and scientific inquiry include the use of highly interactive simulations and collaboration with artificial agents. Advances in ITSs will play an important role in the development of the next generation of assessment systems.

References

1. Self, J.A.: The distinctive characteristics of intelligent tutoring systems research: ITSs care, precisely. Int. J. Artif. Intell. Educ. **10**, 350–364 (1999)
2. Brusilovsky, P., Millán, E.: User models for adaptive hypermedia and adaptive educational systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) The Adaptive Web. LNCS, vol. 4321, pp. 3–53. Springer, Heidelberg (2007)

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, pp. 508–509, 2018.
https://doi.org/10.1007/978-3-319-91464-0

3. Zapata-Rivera, D.: Toward caring assessment systems. In: Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization, pp. 97–100. ACM, New York (2017)
4. Karpicke, J., Blunt, J.R.: Retrieval practice produces more learning than elaborative studying with concept mapping. Science 20 January 2011. http://science.sciencemag.org/content/early/2011/01/19/science.1199327. Accessed 07 April 2018

Making Sense Out of Synchronous and Asynchronous Discourse in Education (SADES)

Nia Dowell[1](✉), Andrew Hampton[2], Xiangen Hu[2], and Christopher Brooks[1]

[1] University of Michigan, Ann Arbor, MI 48108, USA
{Ndowell,Brooksch}@umich.edu
[2] University of Memphis, Memphis, TN 38152, USA
{jhmpton8,Xhu}@memphis.edu

Abstract. This workshop brings together researchers who are interested in theories, technologies, applications, and impacts of synchronous and asynchronous discourse in educational settings (SADES). The last two decades have led to significant changes in education, with digital learning infrastructures such as blended classrooms, computer-mediated collaborative learning environments, intelligent tutoring systems, and most recently massive open online courses (MOOCs). These systems produce an abundance of data streams including natural language, multimedia, and interaction trace data, affording new approaches to educational research. A major advantage of digital learning environments is that researchers have access to the data associated with the full scope of a learner's experience and actions as they navigate through the environment, including the student discussions.

Most of existing ITS applications involve one or at most two interactive conversational avatars (CAs) and one student. However, recent research efforts have been directed towards environments which involve multiple CAs and multiple human learners, which are co-presented in the same interactive intelligent tutoring environment (IITE). In doing so, this work is scaling the interactive elements of more traditional ITSs, and creating opportunities to explore sociocognitive processes in these environments through the use of computational models and natural language interactions.

The majority of automated text analysis systems focus on characterizing the more macro language and discourse properties of an entire batch of texts. That is, they explore phenomena at the student or group level. While certainly useful, few analytical approaches and technological systems allow researchers to explore the micro intra- and interpersonal patterns associated with participants' sociocognitive processes. In this workshop, we highlight recent analytical approaches for exploring both macro and micro SADES processes.

Keywords: Synchronous discourse · Asynchronous discourse
Measurement

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, p. 510, 2018.
https://doi.org/10.1007/978-3-319-91464-0

Optimizing Human Learning

Fabrice Popineau[1(✉)], Michal Valko[2], and Jill-Jênn Vie[3]

[1] LADHAK team, CentraleSupélec/LRI, Orsay, France
fabrice.popineau@lri.fr

[2] SequeL team, INRIA Lille - Nord Europe, Villeneuve-d'Ascq, France
michal.valko@inria.fr

[3] RIKEN Center for Advanced Intelligence Project (AIP), Tokyo, Japan
vie@jill-jenn.net

Abstract. What should we learn next? In this current era where digital access to knowledge is cheap and user attention is expensive, a number of online applications have been developed for learning. These platforms collect a massive amount of data over various profiles, that can be used to improve learning experience: intelligent tutoring systems can infer what activities worked for different types of students in the past and apply this knowledge to instruct new students. In order to learn effectively and efficiently, the experience should be adaptive: the sequence of activities should be tailored to the abilities and needs of each learner, in order to keep them stimulated and avoid boredom, confusion and dropout.

Educational research communities have proposed models that predict mistakes and dropout, in order to detect students that need further instruction. There is now a need to design online systems that continuously learn as data flows, and self-assess their strategies when interacting with new learners. These models have been already deployed in online commercial applications (ex. streaming, advertising, social networks) for optimizing interaction, click-through-rate, or profit. Can we use similar methods to enhance the performance of teaching in order to promote lifetime success? When optimizing human learning, which metrics should be optimized? Learner progress? User addiction? The diversity or coverage of the proposed activities?

Student modeling for optimizing human learning is a rich and complex task that gathers methods from machine learning, educational data mining and psychometrics. This workshop welcomes researchers and practitioners around item response theory, additive/conjunctive factor models, cognitive diagnostic models, (deep) knowledge tracing, models of learning and forgetting, multi-task learning, and brand-new techniques.

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, p. 511, 2018.
https://doi.org/10.1007/978-3-319-91464-0

Industrial Tracks

ITS Adaptive Instruction Systems (AIS) Standards Workshop

Chair

Robert Sottilare (US Army Research Laboratory)

Committee Members

Avron Barr (Aldo Ventures, Inc. and IEEE Learning Technology Standards Committee)
Arthur Graesser (University of Memphis)
Xiangen Hu (University of Memphis)
Keith Brawner (US Army Research Laboratory)
Robby Robson (Eduworks, Inc.)

Summary. This workshop is under the auspices of the 2018 ITS Conference Industry Track and is focused on exploring opportunities for standards for a class of technologies known as Adaptive Instructional Systems (AISs). Adaptive instruction uses computers and AI to tailor training and educational experiences to the goals, learning needs, and preferences of each individual learner and team of learners. Recently, the IEEE Learning Technologies Steering Committee (LTSC) approved the formation of a study group to examine the feasibility and efficacy of standards for AISs. This workshop is intended to expose the broader ITS community to recent activities and plans, and solicit input on low hanging fruit (near-term opportunities) to develop AIS standards.

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, p. 515, 2018.
https://doi.org/10.1007/978-3-319-91464-0

ITS Applications Workshop

Chair

Robert Sottilare (US Army Research Laboratory)

Committee Members

Benjamin Nye (University of Southern California)
Rodney Long (US Army Research Laboratory)
Anne Sinatra (US Army Research Laboratory)
Alan Carlin (Aptima, Inc.)

Summary. This workshop is under the auspices of the 2018 ITS Conference Industry Track and its purpose is to present papers, demonstrate and discuss various ITS applications which use computers and artificial intelligence to tailor instruction (training and educational experiences) to meet the goals, learning needs, and preferences of each individual learner and team of learners. Discussion topics include the design, development, and application of ITS technologies (e.g., learner, pedagogical, domain and interface modeling) and is intended to engage the ITS conference audience in a discussion of their applications, tools, methods, and general experiences with ITSs.

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, p. 516, 2018.
https://doi.org/10.1007/978-3-319-91464-0

Tutorials

Automating Educational Research Through Learning Analytics: Data Balancing and Matching Techniques

David Boulanger(✉), Vivekanandan Kumar, and Shawn Fraser

Athabasca University, Edmonton, AB T5J 3S8, Canada
{dboulanger, vive, shawnf}@athabascau.ca

This tutorial presents guidelines on how to conduct causal analyses in observational study settings and compares the key properties of the gold-standard randomized experiment against the naturally-occurring observational study. It advocates that the randomized experiment is the specific case of the observational study, the general case, where data balance is inherently optimized. This tutorial promotes discussion on the role that learning analytics can play in educational research to enhance causal analysis through the collection of a wider range of digital learning data and the inclusion and participation of a more diverse set of learners.

Although observational studies have garnered considerable interest in past years, they are seen as being not ready yet to decisively overcome randomized experiments. For example, to be accurate, observational studies require identifying as many confounding factors as possible to minimize the underlying bias. Nevertheless, increasing the variety of data types collected and blocking on these variables to approximate randomized block designs without investigating their actual individual and combined causal effects on targeted outcomes constitute a serious threat to their validity by increasing further data imbalance. Hence, observational studies require a holistic approach, where impact of both treatment variables and covariates are simultaneously and iteratively assessed and updated.

Propensity Score Matching became one of the favorite observational methods to investigate naturally occurring data. However, recent literature revealed major weaknesses of PSM such as the data imbalance (PSM Paradox) created by dimensionality reduction and compared alternative approaches like Coarsened Exact Matching and Mahalanobis Distance Matching. It has also been shown that matching techniques may prove to be effective in some scenarios and suboptimal in others, and that several types of matching methods should be tested, including hybrid versions. Several optimization functions then need to be calculated to measure the level of data imbalance in matched control and treatment groups, such as $L1$, Average Mahalanobis Imbalance (AMI), and the average difference in means. An R software package, called MatchingFrontier, has been developed (King et al. 2014) to facilitate the assessment and selection of the best matching methods by means of visualizations. Hence, this tutorial introduces MatchingFrontier and provides directions for further research to create statistical algorithms that will allow the computer to automatically select optimal matching techniques.

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, p. 519, 2018.
https://doi.org/10.1007/978-3-319-91464-0

Authoring, Deploying and Data Analysis of Conversational Intelligent Tutoring Systems

Xiangen Hu[1,2(✉)], Zhiqiang Cai[1], Keith Shubeck[1], Kai-Chih Pai[3], Arthur Graesser[1], Bor-Chen Kuo[3], and Chen-Huei Liao[3]

[1] University of Memphis, Memphis, USA
[2] Central China Normal University, Wuhan, China
[3] National Taichung University of Education, Taichung City, Taiwan

There have been decades of efforts on research and development of intelligent tutoring systems (ITS). ITS assess students' performance from the data collected from the interactions and then adaptively select knowledge objects and pedagogical strategies during the tutoring process to maximize learning effect and minimize learning cost. Delivering content with conversation is always attractive to content authors and students. Research has shown that delivering content through conversation is much more effective than a text. Unfortunately, creating conversational content is difficult. ***First***, in order to have a natural language conversation with a student, the machine has to be able to "understand" the student's natural language input. This involves a research field called "natural language understanding." There isn't a perfect natural language algorithm that can really understand user's free-form speech. ***Secondly***, preparing tutoring speeches for conversations is hard. The essential difficulty is that authors will need to consider the appropriate amount of responses to an infinite possibility of student input. ***Additionally***, it is hard to create and test conversation rules. Conversation rules decide the condition under which a prepared speech is spoken. Since the tutoring conversations often go with other displayed content (e.g., text, image, video) conversation rules need to consider all activity within the learning environment, in addition to the natural language inputs from students. The rule system varies because different environments generate different activity. Creating and testing the rules is also time-consuming. ***We will try to address these issues and introduce some solutions in this one day tutorial***. This tutorial focus on *Authoring, Deploying & Data Analysis of Conversational Intelligent Tutoring Systems*. We use AutoTutor as the demonstrating ITS in this tutorial. AutoTutor is a research-based system framework funded by the US NSF, IES, DoD, Army and Navy. AutoTutor in this tutorial is a collection of ITS that hold conversations with the human in natural language. AutoTutor has produced learning gains across multiple domains (e.g., computer literacy, physics, critical thinking). All AutoTutor implementations have the following important properties: (**1**) they use human-inspired tutoring strategies; (**2**) they use pedagogical agents, and (**3**) they use technologies that support natural language tutoring. At the end of this Tutorial, we expect participants will be able to (**a**) create their own conversational ITS using a web-based authoring tool, (**b**) collect interactive data from their own Conversational ITS and save this data to the standardized database, and (**c**) extract and analyze the data using Datashop.

R. Nkambou et al. (Eds.): ITS 2018, LNCS 10858, p. 520, 2018.
https://doi.org/10.1007/978-3-319-91464-0

Author Index

Abuelsaad, Tamer 392
Afzal, Shazia 267
Aghababyan, Ani 274
Agrawal, Sweety 407
Ahn, Jae-wook 280
Aiemlaor, Nattapon 386
Albacete, Patricia 450
Allain, Allison 392
Alvares, Danilo 180
Arefin, Mohammad Shamsul 446
Arroyo, Ivon 366
Atria, José Joaquín 421
Azevedo, Roger 33, 44, 140, 170, 202, 223, 233, 411, 434

Baierle, Ivan Luis Feix 3
Baker, Ryan S. 130, 274
Balyan, Renu 191
Barbosa, Alexandre de A. 13
Beauchamp, Miriam H. 473
Béland, Sébastien 507
Belaissaoui, Mustapha 425
Beldjoudi, Samia 23
Ben Abdessalem, Hamdi 287, 418, 483
Ben Khedher, Asma 415
Benlamine, Mohamed S. 54, 438
Bevacqua, Elisabetta 470
Biddle, Elizabeth 88
Birk, Gurpreet 442
Biswas, Gautam 233
Bodnar, Stephen 442
Bouchet, François 33, 434
Boukadida, Marwa 287, 418
Boulanger, David 294, 519
Bourdeau, Jacqueline 505
Bourrier, Yannick 300
Boyce, Michael 88
Brooks, Christopher 510
Brito, Patrick H. 13
Bruegge, Bernd 326
Burleson, Winslow 366

Cai, Zhiqiang 520
Chang, Maiga 476
Chaouachi, Maher 438
Chen, Shiping 212
Chen, Su 463
Chen, Yuxin 442
Chen, Zexuan 486
Chounta, Irene-Angelica 450
Cheng, Xiaoxiao 306
Christensen, Heidi 421
Cloude, Elizabeth B. 44
Costa, Evandro de B. 13
Coy, André 421
Cui, Tingru 212
Culbertson, Kirby 411
Cunningham, Stuart 421

da Silva, João Luis Tavares 359
Dang, Nam Hai 280
Dascalu, Mihai 191
Desideri, Lorenzo 421
Desmarais, Michel 507
Dewan, Mohammad Ali Akber 446
Dickler, Rachel 99
Dong, Yi 233
Dorneich, Michael 457
Doumbouya, René 54
Dowell, Nia 510
Dufresne, Aude 54, 438, 473

Eicher, Bobbie Lynn 490
Elkhannoubi, Hasna 425
Emeka, Chinedu 431
Essa, Alfred 130
Essalmi, Fathi 476

Fan, Chia-Chen 399
Fang, Ying 463
Faron Zucker, Catherine 380
Feng, Andrew 256
Fraser, Shawn 519
Frasson, Claude 54, 287, 415, 418, 438
Freedman, Reva 428

Garbay, Catherine 300
Giboin, Alain 380

Gilbert, Stephen 457
Glass, Michael 431
Gluz, João Carlos 3
Gobert, Janice 99
Goel, Ashok 490
Goldberg, Stephen L. 256
Gonzalez-Sanchez, Javier 352
Gottardo, Ernani 313
Graesser, Arthur 520
Graesser, Arthur C. 463
Green, Phil 421
Grubišić, Ani 460
Guo, Jiayu 306
Guo, Qun 306

Haddawy, Peter 386
Hagemann, Nina 319
Haladjian, Juan 326
Hampton, Andrew 510
Hampton, Andrew J. 463
Harley, Jason M. 33, 434, 438
Hassan, Muhammad Mustafa 66
Hayashi, Yugo 78
Hee Kim, Jung 431
Heller, Robert 373
Hmelo-Silver, Cindy E. 442
Hodaie, Zardosht 326
Howe, Andrew 332
Hu, Xiangen 510, 520
Huang, Lingyun 442
Huang, Yazhou 280

Intaraudom, Atirach 386
Iqbal, Ashesh 446
Isotani, Seiji 180
Ito, Takayuki 493

Jagvaral, Yesukhei 431
Jambon, Francis 300
Jang, Eunice Eunhee 202
Jaques, Patricia 180
Jemni, Mohamed 476
Jiao, Jianli 486
Johnson, Amy M. 191
Jordan, Pamela 450
Joyner, David 490
Jraidi, Imène 415

Karabadji, Nour El Islem 23
Katz, Sandra 450

Kazemitabar, Maedeh Assadat 442
Kjallstrom, Jan 280
Kim, Hyeoncheol 466
Kinshuk 476, 499
Kojima, Kazuaki 120
Kokku, Ravi 280, 392
Kumar, Vivekanandan 294, 519
Kuo, Bor-Chen 520

Lajoie, Susanne 110
Lajoie, Susanne P. 442
Lalwani, Amar 407
Lameier, Elizabeth 88
Lau, Clarissa 202
Lester, James 140, 411
Lewkow, Nicholas 274
Li, Deqi 306
Li, Haiying 99
Li, Shan 110
Liao, Chen-Huei 520
Liew, C. W. 339, 454
Limsuvan, Parichat 386
Lin, Fuhua 373
Liu, Xiao 306
Liu, Xiaolong 399
Liu, Yang 438
Lomas, Derek 345
Loye, Nathalie 507
Luengo, Vanda 300
Lusetich, Dennis 450

Matthews, Gerald 88
Malavasi, Massimiliano 421
McCarthy, Kathryn S. 191
McLaren, Bruce M. 450
McNamara, Danielle S. 191
Merchant, Chirag 256
Miglino, Orazio 160
Mizoguchi, Riichiro 505
Miwa, Kazuhisa 120
Mojarad, Shahin 130
Morasse, Frédérick 473
Mojarad, Shirin 130
Moussavi, Raha 99
Mudrick, Nicholas V. 140, 170, 202, 223, 233, 411
Muldner, Kasia 366

Nguyen, H. 339, 454
Nguyen, Phong 332
Nkambou, Roger 473

O'Hara, Sean 280
O'Mahony, Michael P. 319
Okuhara, Shun 493
Ouverson, Kaitlyn 457

Pai, Kai-Chih 520
Patel, Nirmal 345
Paul, Anand 280
Pena, Mariangely Iglesias 457
Penumala, Manohara Rao 352
Pereira, Matheus 359
Peters, Chad 366
Pham, Phuong 150
Poitras, Eric 110
Ponticorvo, Michela 160
Price, Megan J. 140, 170
Procter, Michael 373
Popineau, Fabrice 511
Psyché, Valéry 505

Querrec, Ronan 470
Qureshi, Adnan N. 66

Rajendran, Ramkumar 233
Rega, Angelo 160
Reinerman-Jones, Lauren 88
Reis, Helena 180
Ricardo Pimentel, Andrey 313
Rivera,Veronica 496
Roberts, Candice 140, 411
Robinson, Timothy J. 460
Rodríguez Rocha, Oscar 380
Rudzicz, Frank 421
Ruseti, Stefan 191

Šarić, Ines 460
Savard, Isabelle 505
Sawyer, Robert 140, 411
Schultheis, Holger 386
Schwartz, David 256
Sellman, Collin 345
Sengupta, Bikram 267
Seridi, Hassina 23
Shapiro, Ari 256
Sharma, Aditya 345
Shashidhar, Vinay 267
Shen, Jun 212
Shi, Genghu 463
Sinclair, Jeanne 202
Shubeck, Keith 520
Sindhgatta, Renuka 267
Sipolins, Aldis 280
Smyth, Barry 319
Stankov, Slavomir 460
Suebnukarn, Siriwan 386
Sumi, Kaoru 243
Sun, Chuen-Tsai 399
Sun, Dongeun 466
Sun, Geng 212
Sundararajan, Sharad 280

Taoum, Joanna 470
Tato, Ange 473
Taub, Michelle 44, 170, 202, 223, 233, 411
Taylor, Robert 411
Teemuenvai, Chontee 386
Tejwani, Ravi 280
Terai, Hitoshi 120
Tiam-Lee, Thomas James 243
Tlili, Ahmed 476
Trausan-Matu, Stefan 191

Valko, Michal 511
Vannaprathip, Narumol 386
Vassileva, Julita 508
Vempaty, Aditya 392
Vicari, Rosa Maria 359
Vie, Jill-Jênn 511

Wang, Jingtao 150
Wang, Ning 256
Walton, Jamiahus 457
Woolf, Beverly 366

Xu, Dongming 212

Yan, Hongxin 499
Yuan, Shyan-Ming 399
Yuan, Xiaofang 306

Zapata-Rivera, Diego 508
Zhang, Xuebai 399
Zheng, Juan 110, 442
Zhu, Ruofei 306
Zhu, Zhengzhou 306
Zhuang, Cindy 256