AIED 2013 Workshops Proceedings
Volume 3

2nd Workshop on
Intelligent Support for Learning in Groups
Tuesday, July 9, 2013
Memphis TN

Rohit Kumar
Speech, Language and Multimedia
Raytheon BBN Technologies
USA

Jihie Kim
Information Sciences Institute
University of Southern California
USA

https://sites.google.com/site/islg2013/

Preface

Technological advances in the use of Artificial Intelligence for educational applications over the past two decades have enabled the development of highly effective, deployable learning technologies that support learners across a wide-range of domains and age-groups. Alongside, mass access and adoption of revolutionary communication technologies have made it possible to bridge learners and educators across spatio-temporal divides. On the other hand, research in collaborative learning has informed instructional principles that leverage the pedagogical benefits of learning in groups. Educational service providers including mainstream universities are deploying their courses to online learning platforms that allow students to share their learning experience with their peers. Large volumes of educational content including videos, presentations, books and games are accessible on mobile/tablet devices which enrich learning interactions by bringing students together.

Over the past few years, the AIEd research community has started investigating extension of fundamental techniques (such as student modeling, model-based tutors, integrated assessment, tutorial dialog, automated scaffolding, data mining, pedagogical agents) to support learning in groups. The goal of this series of workshops is to provide a focused forum for bringing this sub-community of AIEd researchers together to share recent advances in the field.

Building on its first instantiation in 2012, this workshop will comprise of presentations describing advances in state of the art AIEd techniques to improve the effectiveness of learning in groups. Five full length papers and six short papers were accepted for presentation this year. These eleven papers are organized into four interrelated areas that cover the breadth of the topics of interest. Additionally, two positions papers accepted to this workshop are included in these proceedings. Besides the paper presentations, the workshop will include a group discussion session. After the workshop, notes from this session, will be shared on the workshop website.

June, 2013
Rohit Kumar, Jihie Kim

Organizers

Jihie Kim	*University of Southern California, USA* (jihie@isi.edu)
Rohit Kumar	*Raytheon BBN Technologies, USA* (rkumar@bbn.com)

Advisors

Arthur C. Graesser	*University of Memphis, USA*
Lewis W. Johnson	*Alelo Inc., USA*
James C. Lester	*North Carolina State University, USA*
Carolyn P. Rosé	*Carnegie Mellon University, USA*
Beverly P. Woolf	*University of Massachusetts Amherst, USA*

Program Committee

Ari Bader-Natal	*Minerva Project, USA*
Ryan S.J.d Baker	*Columbia University, USA*
Kristy Boyer	*North Carolina State University, USA*
Stavros N. Demetriadis	*Aristotle University of Thessaloniki, Greece*
Toby Dragon	*Saarland University, Germany*
Gahgene Gweon	*Korea Advanced Institute of Science and Technology, S. Korea*
Sharon I-Han Hsiao	*Columbia University, USA*
Charalampos Karagiannidis	*University of Thessaly, Greece*
Judy Kay	*University of Sydney, Australia*
Fazel Keshtkar	*University of Memphis, USA*
Diane Litman	*University of Pittsburgh, USA*
Bruce M. McLaren	*Carnegie Mellon University, USA* *Saarland University, Germany*
Elaine Raybourn	*Sandia National Labs, USA*
Oliver Scheuer	*Saarland University, Germany*
Erin Walker	*Arizona State University, USA*

Additional Reviewers

Tiffany Barnes	*University of North Carolina at Charlotte, USA*

Table of Contents

Position Papers

Authoring Collaborative Intelligent Tutoring Systems

Jennifer K. Olsen[1], Daniel M. Belenky[1], Vincent Aleven[1], Nikol Rummel[12], Jonathan Sewall[1], and Michael Ringenberg[1],

[1]Human Computer Interaction Institute, Carnegie Mellon University, Pittsburgh, PA, USA
jkolsen@cs.cmu.edu, dbelenky@andrew.cmu.edu,
{aleven,sewall,mringenb}@cs.cmu.edu
[2]Institute of Educational Research, Ruhr-Universität Bochum, Germany
nikol.rummel@rub.de

Abstract. Authoring tools for Intelligent Tutoring System (ITS) have been shown to decrease the amount of time that it takes to develop an ITS. However, most of these tools currently do not extend to collaborative ITSs. In this paper, we illustrate an extension to the Cognitive Tutor Authoring Tools (CTAT) to allow for development of collaborative ITSs that can support a range of collaboration scripts. Authoring tools for collaborative ITSs must be flexible enough to allow for different learning goals and different collaboration scripts. We discuss how two collaboration scripts that we are using in our research on fractions learning are implemented in CTAT. The examples illustrate how CTAT flexibly supports collaborative tutors by running synchronized tutor engines for each student, and how it supports the development of collaborative tutors through the use of multiple behavior graphs that use no programming to develop.

Keywords: Problem solving, collaborative learning, intelligent tutoring system, authoring tools

1 Introduction

Collaborative learning has been shown to be effective for student's knowledge acquisition in some computer-supported settings [9]. However, there is a lack of effective and flexible authoring tools for collaborative learning activities. Authoring tools for Intelligent Tutoring Systems (ITSs) are often geared towards *individual* learning and typically do not have support for the components that make collaborative learning effective [11]. Within Computer Supported Collaborative Learning, collaboration scripts are often used to support collaborative learning, but are often either developed specifically for a particular application [8] [17] or, at best, are provided through a tool that can be used for reuse of the *same* script across multiple subject areas [1], [3], [7], [10], [13-14], [16]. In both approaches, the development tailored for particular domains and learning goals is not straightforward and may not even be feasible. A tool that can be used to flexibly author a range of collaboration scripts for a range of subject areas would bridge this gap. We are working on creating such a tool, by extending an existing ITS authoring tool, the Cognitive Tutor Authoring Tools (CTAT) [2], so it aids in the development of tutors that integrate a range of collaboration scripts. An earlier attempt to extend CTAT [4] focused on log data, not scripting.

Collaboration scripts are used to structure the tasks and interactions within a group. According to Kollar, Fischer, and Hesse [6], a collaboration script within the educational domain consists of at least five components: the learning objectives, the types of learning activities, the sequencing of the activities, role distribution, and how the script is represented. These components are a way to compare collaboration scripts across platforms, such as face-to-face and computer-supported settings and provide a guideline for the coverage that is needed in authoring tools that wish to support collaborative learning.

There has been work to make collaboration scripts generalizable across learning domains. One example of an authoring tool that can be used across different learning domains is the work done with conversational agents, which monitor a group conversation and can intervene when needed [1], [7]. Although this authoring tool supports multiple learning domains, it supports only the development of collaboration scripts that rely on the use of conversational agents and not a more general class of collaboration scripts. Other tools aim to reuse existing collaboration scripts for new scenarios [3], [10], [13], [16]. These tools are dependent on the learning goals that the existing collaboration script supports instead of customizing the collaboration script for the desired learning goals. On the other hand, the tool, XSS, which is a framework for rapidly developing computer-supported collaboration scripts for new technologies, does support the creation of collaboration scripts to meet specific learning goals [14]. However, XSS does not have support for authoring scripts through an interface, so it may be difficult for users with less programming experience.

The enhancement to CTAT described in this paper allows authoring of collaborative ITSs without programming, and the collaboration script can be specific to the learning goals of the tutor being developed. In this paper we provide collaboration script examples that support cognitive group awareness [4] and sharing of unique information, illustrating the flexibility of the CTAT authoring tool for collaboration. The enhancement to the CTAT system allow students to collaborate through synchronized tutor engines and we will describe how it supports collaborative tutor problems.

2 Collaboration Examples Using CTAT for Collaboration

2.1 An Example of Support for Cognitive Group Awareness

Before we describe how we modified CTAT so it supports authoring of collaborative tutoring, we describe two examples of collaborative tutoring behavior authored with this tool. Specifically, building on our prior work on the Fractions Tutor [12], we are creating a collaborative tutoring system to help elementary students learn fractions. The current prototype includes four conceptual problems and four procedural problems focused on equivalent fractions, each with embedded collaboration scripts. The prototype tutor has been pilot tested with four dyads so far. As students use the tutor, they talk to each other via Skype. The two examples illustrate the types of collaboration scripts can be implemented using the collaborative version of CTAT. In the next section, we extended CTAT to support the collaborative features of these tutors.

The first example features a collaborative fractions problem with a script that supports cognitive group awareness, in which the student is learning conceptual knowledge about equivalent fractions. Cognitive group awareness is the awareness that comes from having information about group members' knowledge, information, or opinions and has been shown to be effective for the collaboration process [5]. This awareness can be supported through tools such as skill meters or by using an interactive interface to display a partner's answers. In our tutor, cognitive group awareness during problem solving is structured as follows: First, the collaborating partners each answer the same question separately. The tutor then displays both partners' answers to promote discussion, and the partners provide a final answer endorsed by both. Each student is given a pair of contrasting attributes (see Figure 1, panel B2) about the fractions. The students are not given feedback on their individual answer but are shown what their partner selected. This allows each student to see their partner's understanding of the fractions. The students are then asked to discuss their answers and decide as a pair what the correct answer will be. Having each student display his or her knowledge of the given fractions before discussing the question together supports the cognitive group awareness. This discussion can lead to a mutual understanding of the fraction attributes, which supports a better understanding of the conceptual knowledge for equivalent fractions. As may be clear, to support cognitive group awareness, the collaborative tutor provides different views of the same problem to the collaborating partners, using two synchronized tutor engines as described below.

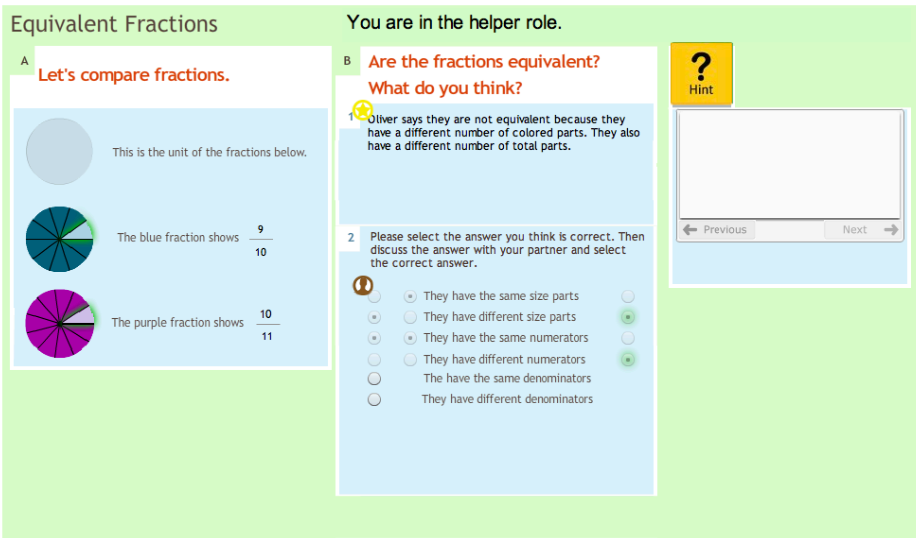


Fig. 1. Panel B2 displays an example of support for cognitive group awareness through the use of multiple radio buttons where each student first selects an answer based on their knowledge before the group makes a group selection that is tutored.

2.2 An Example of Support for Sharing Unique Information

We also used the collaborative version of CTAT to implement a second type of fractions problem, in which students learn how to procedurally evaluate equivalent fractions. As in the previous example, the collaborative tutor provides a different view on the same problem for each collaborating partner, although this time the collaboration is scripted differently for the different learning objective. Specifically, we implemented a script that distributes unique information between the partners and supports the sharing of this information. Students are shown a fraction expressed in symbols (see Figure 2) that their partner does not see as indicated by the star icon. Each partner is also given a circle diagram that they can interact with; their partner can see this diagram but cannot interact with it as indicated by the silhouette icon. One student is first asked to share their fraction with their partner (i.e., by telling their partner about it) while the second student is asked to make this fraction using their circle diagram. The students then switch roles and one student shares their fraction while the other student makes this fraction. Each student sees the feedback from the tutor, so if a student is struggling to correctly make the fraction, their partner, who can see the fraction and the tutor feedback, can provide support and help. By providing each student with different information, the students need to start a dialogue and share. This activity makes the students aware of the fractions as a first step to supporting procedural knowledge for evaluating equivalent fractions.

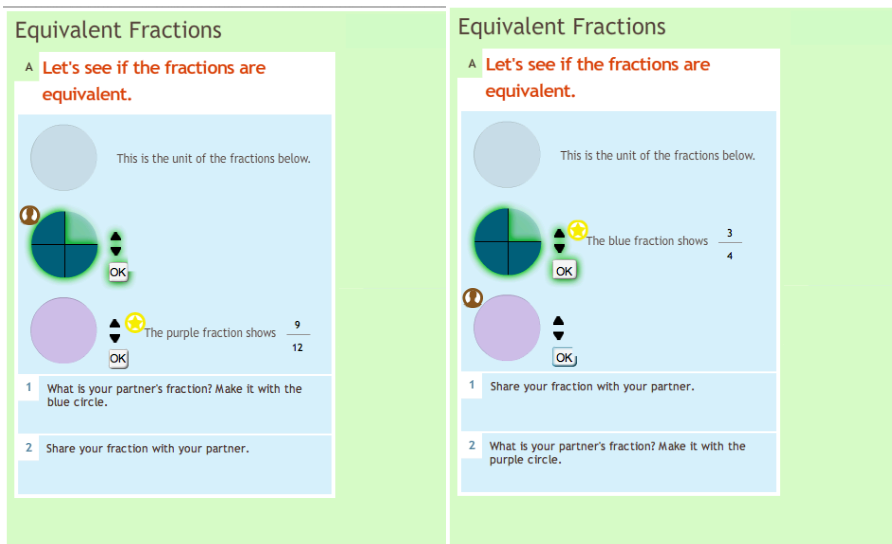


Fig. 2. Panel A displays an example of individual information that needs to be shared between participants. The top blue fraction was made by the student on the left screen using the information shared by the student on the right screen. The purple fraction will be made with the student on the right screen with the information from the student on the left screen.

Both examples illustrate a range of collaborative activities that can be supported using CTAT for collaboration. Kollar, Fischer, and Hesse specify collaboration scripts by focusing on five components [6]. These five attributes provide a guideline for the coverage that is needed in authoring tools. Both examples use different *learning activities* to support the learning goals of the problems. The sharing of unique information uses activities such as sharing and problem solving where as the script that supports cognitive group awareness uses activities such as sharing knowledge and mutual explanations. Within these activities the students are also assigned to very different *roles* where in the unique information scenario they are asked to be a sharer or to be a problem solver and then switch roles. In the support for cognitive group awareness, both students are responsible for sharing their knowledge and then discussing the answers.

3 Authoring Tool Extensions to Support Collaboration

Until recently CTAT only supported tutors for individual use. We focus on one type of tutor that can be authored with CTAT, namely, example-tracing tutors [2]. To develop such a tutor, an author creates two key components, both without programming: a user interface designed specifically for the problem type being tutored (the interface lays out the problem steps) and a generalized behavior graph, which stores all of the acceptable solution paths along with commonly-occurring incorrect steps. The tutor uses the behavior graph to monitor student problem solving and provide guidance to students. Each behavior graphs consists of a set of links that correspond to steps that can be taken in the problem, such as typing in the numerator to a fraction. Some steps (explicitly marked as such) represent *tutor-performed actions*, such as showing a component in the tutor interface that was hidden before. To evaluate student input, the tutor compares the student's problem-solving steps against those in the behavior graph, testing whether the student is on one of the paths in the graph. An author may specify constraints on the order of steps. Behaviorally, example-tracing tutors are similar to other types of ITSs, providing all the key functionality singled out by VanLehn [15] as typical of ITSs.

3.1 Authoring Collaborative Tutors

To expand CTAT so it supports *collaborative example-tracing tutors*, we added the capability to run *multiple synchronized tutor engines,* one for each student in a collaborating group. This set up allows for great flexibility in authoring tutors with embedded collaboration scripts. Specifically, each student in a group has their own behavior graph file and interface file for the given problem. The collaborative version of CTAT synchronizes the tutors so that when one of the collaborating students takes an action, this input is sent to both that student's tutor engine and their partner's tutor engine. Similarly, tutor output is shared among the members of a collaborating group (i.e., all output from the two synched tutor engines, such as hints and feedback, is sent to each student interface separately). One result of this output sharing is that student actions taken on one interface will be "mirrored" on the other interface in the corresponding interface component, together with the associated tutor feedback. As we extended CTAT, we updated the interface tool components to include new actions that better support collaborative learning activities. As an example, we updated the existing components to allow students to view the options of a component without being able to take action on the component, as illustrated in the examples above. We are also adding a highlighting functionality so each student can easily reference a component.

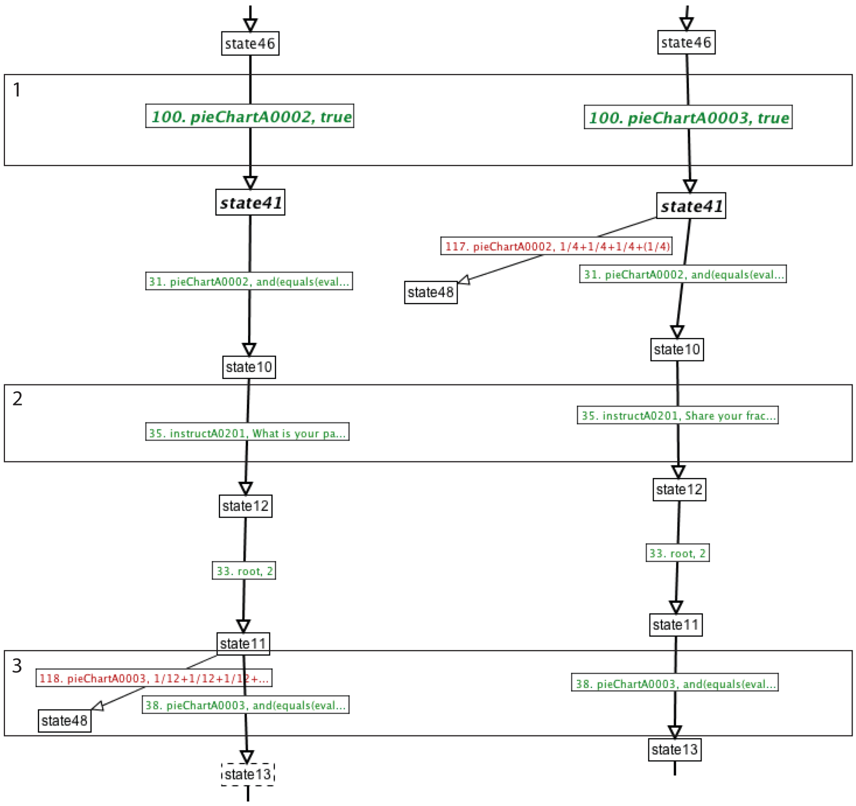


Fig. 3. Excerpts from two behavior graphs working for a single problem. Together both behavior graphs capture the first step to be completed by the students for the problem in Fig 2. Box 1 demonstrates the different locking of components for each student, Box 2 demonstrates different instructions for each student, and Box 3 demonstrates the use of student-performed actions to advance the state of the problem where the partnering student can only take the action.

With these collaborative extensions to CTAT, an author can create tutors that do not differ for the collaborating partners - simply by supplying the same behavior graph and interface for each collaborating partner. The result would be a tutor with which two students interact simultaneously and synchronously while each sitting at their own computer. They would each see the changes that their partner makes. This kind of collaboration may not be terribly useful, however. The power of the approach comes from being able to craft tutors in which the collaborators have different views on the same problem and have different sets of actions available to them. There are many collaboration activities, such as the jigsaw and the tutee/tutor paradigm, where the benefit of the activity comes from the students having different roles and responsibilities in the problem-solving task. The CTAT authoring tool supports this kind of differentiation, as an author can create separate behavior graphs, one for each student, that display different instructions or capture different student problem-solving actions, dependent on the role of each student, as is used in the cognitive awareness activity. For example, Figure 3 shows, side-by-side, two behavior graphs for the support of unique information example illustrated in Figure 2. These two behavior graphs share common structure, but also differ so as to support different interactions for the two collaborating students.

To show different instructions for each student, an author can use a different tutor-performed action at the corresponding link in the two behavior graphs. An example is shown in Box 2 of Figure 3 where each student receives different directions from the behavior graph at the same point in time. (The label on the link shows the message displayed to the student in truncated form.) Similarly, by providing different behavior graphs for each member, the actions taken by the users can differ. One way to make different sets of actions available to each collaborating partner is by locking certain components in the interface, a different set for each partner. This allows both students to see the action on their respective interfaces while only allowing one student to be able to take the action. An example is shown in Box 1 of Figure 1 where different components (the two circle components, pieChartA0002 and pieChartA0003) are locked for the students through a tutor-performed action, preventing them from interacting with that component. The result of this link in the behavior graph is seen in Figure 2 where the circle that corresponds to the fraction shown on the screen is locked for that student, so that each student can perform his/her own role but not his/her partner's role. Though the student cannot act on the component that is locked, a step to solve the component is in the behavior graph (see Box 3 of Figure 3) so that the problem will not advance until their partner has completed the step. An author can also make the tutor accept different actions from each student by recording different actions in each student's behavior graph. In this case, the student without the action recorded would not have to wait for this action to take place to continue working on the next step of the problem.

Another way to provide different interface elements to the members of each dyad is through an interface file. This file is a SWF file created in Flash. The author can select the components, control their placement on the interface, set basic parameters, and use custom code if necessary. In this way, an author can tailor the interface for the different roles that the collaborators have in the collaboration script that is being supported. An author can also determine what feedback each student receives during the problem by setting an initial tutor feedback parameter for each interface component. This parameter controls whether or not there will be tutor feedback on actions on that component. For example, in the cognitive awareness task in Figure 1, the radio buttons that correspond to the student's *individual* answers provide no feedback, as they serve mainly to support the partners' mutual awareness of each other's reasoning. On the other hand, the radio buttons for the *group* answer (on the right in Figure 1) provide correct or incorrect feedback.

The steps to develop a tutor using CTAT consist of developing a user interface, creating a behavior graph, and annotating the behavior graph [2]. Within CTAT, an interface is built using an interface builder and the different components of the interface are adding using a drag-and-drop method. Each component has a set of parameters that can be set allowing the developer to customize the look and feel of the parameter to match their tutor layout. This allows a developer to create a tutor interface without the need for any coding on the part of the developer. Once an interface is created, a behavior graph can be created that maps out the tutor steps through correct and incorrect actions. The behavior graph can be created through demonstrating the actions to be taken on the interface. While having the CTAT Behavior Recorder in demonstration mode, any action that is taken on the interface will be recorded on a behavior graph. By starting at different points in the behavior graph, different branches can be created. This allows a developer to create a behavior graph without the need of programming. After the behavior graph is created through demonstration, the graph can be annotated. Annotation includes adding hints to the links and identifying knowledge components.

To author a collaborative tutor each of the steps to create an individual tutor are followed for each member of the collaboration. Depending on the type of collaboration activities and roles depends on if different tutor interfaces and behavior graphs need to be made for each student in the group or if the same files can be used. If the students are going to be seeing something visually similar then the same tutor interface can be used. If the students are going to take the same actions during the problem, then the same behavior graph can be used. When developing a collaborative tutor, if different interfaces are going to be used and an action that one student takes should be reflected in the view to the other students, then the components that are used for this activity need to be named the same in both interfaces. This is shown in Box 3 of Figure 3 where the same component name is referenced in both behavior graphs. This allows the tutors to reflect an action taken on one interface on the other interface as well. On the other hand, if the author wants particular actions within a tutor interface to be private to one of the collaborating students, one way to do so is to not provide a corresponding interface component in the interface for the other student. The enhancements to CTAT did not add a need for a developer to program to create a tutor. Currently, to test a collaborative tutor, the tutor must be run through the tutoring service. A different browser window can be opened for each student interface so the actions can be seen simultaneously. By assigning each interface and behavior graph to a “student” and then identifying those students as being in a class together, the different tutors are synced and allows communication between the tutors. This assignment can be done through filling out fields in a user interface and no special programming is needed on the part of the author.

4 Discussion, Conclusion and Future Work

Computer-supported collaboration has been shown to be an effective learning paradigm for knowledge acquisition [9], yet most tools that support collaboration do not allow for the authoring of a range of collaboration scripts. Authoring tools for ITSs have been used to address a wide range of domains, but we are not aware of any that support collaboration scripts, other than an early attempt to extend CTAT [4] so it builds collaborative tutors from log data. We extended CTAT so it supports the authoring of collaborative tutors while maintaining its advantages for individual tutoring without programming. With this new version of CTAT, authors can develop collaborative ITSs to meet a range of domains and collaboration scripts. The developer does not need to have a strong background in programming to make a functional tutor.

The extension to CTAT allows for a range of collaboration scripts to be developed. Two examples were provided in this paper, but we also created tailored collaboration scripts to match the learning objectives of six other fractions problems. The flexibility to develop these scripts is because the collaborative version of CTAT was not developed to implement a specific script but to remain open-ended. This design also allows flexibility while developing tutors. As we continue to develop our collaborative fractions tutor, we are taking an iterative approach in which we repeatedly test the collaboration script with students and then refine it to best support the learning goals based on the outcomes of the pilot studies. The collaborative version of CTAT allows for these changes to be made easily in a problem, as behavior graphs are relatively malleable.

Future work will consist of extending CTAT so it can support more than two students in the group. Other future work will be to allow the specifying of groups at runtime instead of needing to specify groups ahead of time. By being able to specify the members of a group at runtime, there would be more flexibility in grouping students in a classroom on any given day. Students would not be dependent on their partner also being there that day. Also to improve the authoring process, functionality is being added to allow an author to have multiple behavior graphs open so they can compare the steps and can copy and paste steps from one graph to another that are similar. The eventual goal of our project is to investigate how best to combine individual and collaborative modes of learning.

Acknowledgments. We thank the Cognitive Tutor Authoring Tools team for their help. This work was supported by Graduate Training Grant # R305B090023 and by Award # R305A120734 both from the US Department of Education (IES).

5 References

1. Adamson, D., & Rosé, C. P.: Coordinating Multi-Dimensional Support in Collaborative Conversational Agents. In: Intelligent Tutoring Systems (pp. 346-351). Springer Berlin Heidelberg (2012)

2. Aleven, V., McLaren, B. M., Sewall, J., & Koedinger, K. R.: A New Paradigm for Intelligent Tutoring Systems: Example-tracing Tutors. International Journal of Artificial Intelligence in Education, 19, 105-154 (2009)
3. Harrer, A., Malzahn, N., & Wichmann, A.: The remote Control Approach-An architecture for Adaptive Scripting Across Collaborative Learning Environments. Journal of Universal Computer Science, 14(1), 148-173 (2008)
4. Harrer, A., McLaren, B. M., Walker, E., Bollen, L., & Sewall, J.: Creating cognitive tutors for collaborative learning: Steps toward realization. User Modeling and User-Adapted Interaction, 16(3-4), 175-209 (2006)
5. Janssen, J., & Bodemer, D.: Coordinated Computer-Supported Collaborative Learning: Awareness and Awareness Tools. Educational Psychologist, 48(1), 40-55 (2013)
6. Kollar, I., Fischer, F., & Hesse, F. W.: Collaboration scripts–a conceptual analysis. Educational Psychology Review, 18(2), 159-185 (2006)
7. Kumar, R., Rosé, C. P., Wang, Y., Joshi, M., & Robinson, A.: Tutorial dialogue as adaptive collaborative learning support. Frontiers in Artificial Intelligence and Applications, 158, 383 (2007)
8. Lesgold, A., Katz, S., Greenberg, L., Hughes, E., & Eggan, G.: Extensions of intelligent tutoring paradigms to support collaborative learning. In S. Dijkstra, H. P. M. Krammer, & J. J. G. van Merrienboer (Eds.), Instructional models in computer-based learning environments. (pp. 291-311). Berlin: Springer-Verlag (1992)
9. Lou, Y., Abrami, P. C., & d'Apollonia, S.: Small group and individual learning with technology: A meta-analysis. Review of educational research, 71(3), 449-521 (2001)
10. Miao, Y., Hoeksema, K., Hoppe, H. U., & Harrer, A.: CSCL Scripts: Modelling Features and Potential Use. In Proceedings of the 2005 conference on Computer support for collaborative learning: learning 2005: the next 10 years! (pp. 423-432). ISLS (2005)
11. Murray, T., Blessing, S., & Ainsworth, S.: Authoring tools for advanced technology learning environments: Toward cost-effective adaptive, interactive and intelligent educational software. Amsterdam: Kluwer Academic Publishers (2003)
12. Rau, M., Aleven, V., Rummel, N., & Rohrbach, S.: Sense Making Alone Doesn't Do It: Fluency Matters Too! ITS Support for Robust Learning with Multiple Representations. In: Intelligent Tutoring Systems, pp. 174-184. Springer Berlin/Heidelberg (2012)
13. Ronen, M., Kohen-Vacs, D., & Raz-Fogel, N.: Adopt & Adapt: Structuring, Sharing and Reusing Asynchronous Collaborative Pedagogy. In Proceedings of the 7th International Conference on Learning Sciences (pp. 599-605). ISLS (2006)
14. Stegmann, K., Streng, S., Halbinger, M., Koch, J., Fischer, F., & Hußmann, H.: eXtremely Simple Scripting (XSS): A framework to speed up the development of computer-supported collaboration scripts. In Proceedings of the 9th International Conference on Computer supported Collaborative Learning-Volume 2 (pp. 195-197). ISLS (2009)
15. VanLehn, K.: The Relative Effectiveness of Human Tutoring, Intelligent Tutoring Systems, and Other Tutoring Systems. Educational Psychologist, 46, 197-221, (2011)
16. Wecker, C., Stegmann, K., Bernstein, F., Huber, M. J., Kalus, G., Rathmayer, S., Kollar, I., & Fischer, F.: Sustainable script and scaffold development for collaboration on varying web content: the S-COL technological approach. In Proceedings of the 9th international conference on Computer supported collaborative learning-Volume 1 (pp. 512-516). ISLS (2009)
17. Walker, E., Rummel, N., & Koedinger, K.: CTRL: A research framework for providing adaptive collaborative learning support. User Modeling and User-Adapted Interaction: The Journal of Personalization Research (UMUAI), 19(5), 387-431 (2009)

A Data Mining Approach to Construct Production Rules in an Educational Game

Fazel Keshtkar, Borham Samei, Brent Morgan, and Arthur C. Graesser

University of Memphis
Institute for Intelligent Systems
Memphis, TN, USA.
{fkshtkar,bsamei,brent.morgan,a-graesser}@memphis.edu

Abstract. One of the most crucial aspects of Intelligent Tutoring Systems in a collaborative serious game is production rules. Given the large number of interactions and conversation between players, it is difficult to follow student questions and reactions in the game environment. Therefore, creating a sophisticated method to construct production rules for handle the students' interactions will boost the performance of the system. In this paper, we propose a state-of-the-art computational approach to automatically generate production rules using co-occurrences of distinct terms from a corpus of students' conversations. Moreover, our model is able to generate additional production rules as new data is available. Finally, we also introduce how to transfer extracted co-occurrences into production rules, and how to build these into the game system.

Keywords: Intelligent Tutoring Systems, Production Rules, Data Mining

1 Introduction

Serious games are increasingly becoming a popular, effective supplement to standard classroom instruction [9]. Some classes of serious games provide microwords [7] that allow players to explore a virtual environment. These simulations have ideal and often simple problems with targeted scaffolding to help users identify important concepts and think critically about them. Multi-party chat is pervasive in recreational games and often crucial to success in multi-player epistemic games [4, 3, 8]. In this paper, we present a method of production rule We employed a computational approach to determine the critical features of multi-party chat in a serious game. We analyzed a corpus of chat conversations and high frequency features along with their co-occurrences. We describe the resulting model below, as well as the process of generating production rules. Finally, we discuss how to utilize this model in the context of a serious game to provide relevant suggestions to a human mentor.

2 Production Rules

A Production Rule consists of a collection of $IF...THEN$ rules that together form an information processing model of some task, or range of tasks. Each rule has two parts: a condition part and an action part. Production rules can be represented in various forms [2], e.g.: “IF condition THEN action”, “IF premise THEN conclusion” or on the other hand “IF proposition p_1 and proposition p_2 are true THEN proposition p_3 is true”. In the context of a serious game, for example, it is likely that the players will eventually need help navigating the user interface. Whereas they would normally ask a human mentor to guide them, if a relevant production rule is built in the system, this situation can easily be detected and resolved by the system, saving the resources of the human mentor. The system outlined below must be able to detect the specific facts or features (such as “email” and “check”) to specify relevant conditions and return the appropriate suggestion. As a result, a computational data mining approach helped us to extract these conditions and facts.

2.1 Speech Act Classification

We selected a system for classifying speech acts [5]. Analyzes of a variety of corpora, including chat and multiparty games, have converged on a set of speech act categories that are both theoretically justified and that also can be reliably coded by trained judges [6]. Our classification scheme has 8 broad categories: **Statements** are verbal reports on scientific facts. **Requests** include asking other participants in the conversation to provide information. **Questions** are queries for information from the addressee. **Reactions** are short verbal responses to requests or questions. **Expressive Evaluations** consist of feedback regarding the player’s performance. **MetaStatements** are statements about the communication process. **Greetings** are expressions regarding any party’s entrance to. **Other** represents speech acts which did not fit into the above categories.

2.2 Land Science Game

Urban Science is an epistemic game created by education researchers at the University of Wisconsin-Madison, designed to simulate an urban planning practicum experience [1]. Young people role-play as professional urban planners in an ecologically-rich neighborhood. The players’ primary task is to redesign the city of Lowell, Massachusetts. Players are assigned to one of three planning teams, and interact with team members and a human mentor using a group chat interface [4, 3, 8]. The ”Question” category is likely the most critical speech act when it comes to addressing player problems. We analyzed 26720 unique chat turns across three instances of Land Science data set.

3 Our Approach

In our model, we identify the relevant facts needed to satisfy the conditions in IF ... $THEN$. Based on these facts, we are able to generate suggestions for a human mentor. In our algorithm, facts can have any of the following features: words, tokens, event, status of the game, or patterns of player’s conversation.

Table 1. shows some of tokens that have high co-occurrence

Token 1	Token 2	co-occ	categories	rooms
stakeholders	what	413	Statement Question Request Reaction	7 5 3 4 12
email	maggie	353	Statement Request Question ExpressiveEval	3 2 4 5 6
want	what	306	Statement Request Question Reaction	5 7 3 4 12
meeting	team	281	Statement Request Question ExpressiveEval	4 3 2 10 11
out	what	280	Statement Request Question Reaction	7 4 3 5 2
now	what	262	Statement Question Request Reaction	7 3 2 10 1
find	out	257	Statement Question Reaction Request	5 10 4 3 7
final	proposal	237	Statement Reaction Request Question	12 2 11 3 6
preference	survey	236	Statement Request Reaction Question	6 9 7 5 10
stakeholders	want	229	Statement Request Question Reaction	5 7 4 3 12

Using these facts, we can generate production rules which offer suggestions for a human mentor.

3.1 Computing Co-Occurrences

One of the most important features to build production rules based on a data-mining approach is to determine the co-occurrences of high or even low frequency tokens in the corpus. In the following sections we describe these features and we show how they can be considered as conditions and facts in our production rules. After preprocessing the corpus, we split each utterance into tokens using the OpenNLP tokenizer, a Natural Language Processing Java Library. We used standard stop words to remove unnecessary tokens. We computed the frequency of all remaining tokens in the corpus for each Speech Act category. These tokens are based on Unigram Entropy Cues and Speech Act classification method that developed by [5]. Then, we ranked these frequencies list from high to low order. In addition to token frequency, it is also critical to assess the relevance of each token, as it may be context-specific. We assessed token relevance by computing co-occurrences. Table 1 shows some examples of co-occurrences in our corpus. In Table 1 tokens that have high co-occurrence chance along with the categories and rooms they appeared in. The categories and rooms are ordered by the frequency of the co-occurrence.

3.2 Constructing Production Rules

As we described in previous sections, Production Rules are in forms of IF ... $THEN$ statements. These IF ... $THEN$ statements must obtained by the **Conditions** and the **Facts** to achieve some **Conclusions** or **Actions**. By looking at Table 1, we see the co-occurrences for “Virtual” are: navigation, stakeholder, neighborhood, character, site, during, etc. In our model, we assume that the facts for conditions can be one or more of the co-occurrences for each token.

4 Conclusion and Future Work

In this paper, we discussed the concept of production rules. These rules are $IF...THEN$ statements which contain some conditions (based on relevant facts). When conditions are met, they trigger some system response, such as a suggestion to players from a mentor or intelligent agent. We introduced a state-of-the-art data-mining approach to construct production rules from a corpus of chat conversations. For future work, we plan to use rule based model to generate production rule. This will allow us to fire relevant functions to produce better suggestions. We also plan to analyze more data to construct additional production rules for the Land Science.

Acknowledgments

This work was funded by the National Science Foundation (DRK-12-0918409).

References

1. Bagley, E.: Stop Talking and Type: Mentoring in a Virtual and Face-to-Face Environmental Education Environment. Ph.D. thesis, University of Wisconsin-Madison (2011)
2. Compton, P., J.R.: A philosophical basis for knowledge acquisition. knowledge acquisition 1990; 2: 241-257
3. D. Ketelhut, C. Dede, J.C.B.N.C.B.: Studying situated learning in a multi-user virtual environment. In: Baker, E., Dickieson, J., Wul-feck, W., ONeil, H. (eds.) Assessment of problem solving using simulations, pp. 37-58. Earlbaum, Mahweh (2007)
4. Dieterle, E., Clarke, J.: Multiuser virtual environments for teaching and learning. In: Pagani, M. (ed.) Encyclopedia of multimedia technology and networking (2nd ed). Idea Group, Hershey 2012.
5. Moldovan, C., Rus, V., Graesser, A.C.: Automated speech act classification for online chat. In: Procceding of FLAIRS 2011 (2011)
6. R.G. DAndrade, M.W.: Speech act theory in quantitative research on inter-personal behavior. In: Discourse Processes. 8 (2), 229-259 (1985)
7. Rickel, J., Lesh, N., Rich, C., Sidner, C., Gertner, A.: Building a bridge between intelligent tutoring and collaborative dialogue systems. In: Proceedings of the 10th Int. Conf. on Artificial Intelligence in Education, San Antonio, TX, pp. 592-594, May 2001.
8. Shaffer, D.: How Computer Games Help Children Learn. Palgrave, New York (2007)
9. Shaffer, D., Chesler, N., Arastoopour, G., DAngelo, S.: Nephrotex: Teaching first year students how to think like engineers. Laboratory Improvement (CCLI) PI Conference (2011)

Modeling the Process of Online Q&A Discussions using a Dialogue State Model

Shitian Shen and Jihie Kim

University of Southern California Information Sciences Institute

4676 Admiralty Way, Marina del Rey, CA, U.S.A

shitians@usc.edu, jihie@isi.edu

Abstract:

Online discussion board has become increasingly popular in higher ed-ucation. As a step towards analyzing the role that students and instructors play during the discussion process and assessing students' learning from discussions, we model different types of contributions made by instructors and students with a dialogue-state model. By analyzing frequent Q&A discussion patterns, we have developed a graphic model of dialogue states that captures the information role that each message plays, and used the model in analyzing student discussions. We present several viable approaches including CRF, SVM, and decision tree for the state classification. Using the state information, we analyze information exchange patterns and resolvedness of the discussion. Such analyses can give us a new insight on how students interact in online discussions and kind of assistance needed by the students.

Keywords: online discussions, dialogue transition, speech act, CRF.

1. Introduction

Online discussion boards, an application of social network on education, provides a platform for students and instructors to share their ideas or to discuss their question not only in traditional courses but also in web-based courses. Such tools can help students solve their problems opportunely, as well as improving instructors' work efficiency. As the discussion board usage increases, we want to understand how students interact with instructors and peers, and how they learn through that interaction.

Although research in online chat and discussion analysis has been increasing re-cently [8,12,14], there has been limited research on modeling the process of information exchange in Q&A forums or how resolvedness of discussions can be determined. In order to analyze and model the process of information exchange, we map interactions in discussions into a Q&A dialogue state model. The state for each message illustrates the status and function of the given message in the Q&A process (discussion thread) [5,6]. We identified six distinctive and frequent states in the discussion process: Problem presenting, Problem understanding, Solving, Solution understanding, Solution objecting, and Solution appreciation. In order to classify the dialogue states efficiently, we apply machine classifiers including linear Conditional Random Fields (CRF), a widely used tool for characterizing the sequential data. The features are generated from message content and positional information, including cue word posi-tions, participants' order, which provides additional hints for state labeling.

The results indicate that frequent states can be reasonably identified using ma-chine classifiers. We demonstrate that the state model can be used in finding frequent patterns in the dialogue progress and evaluating the roles the instructor and students play during the Q&A discussion process. Furthermore, we show that state information can help identifying unresolved discussions, which can be reported to the instructor.

2. State transition model of Q&A discussions

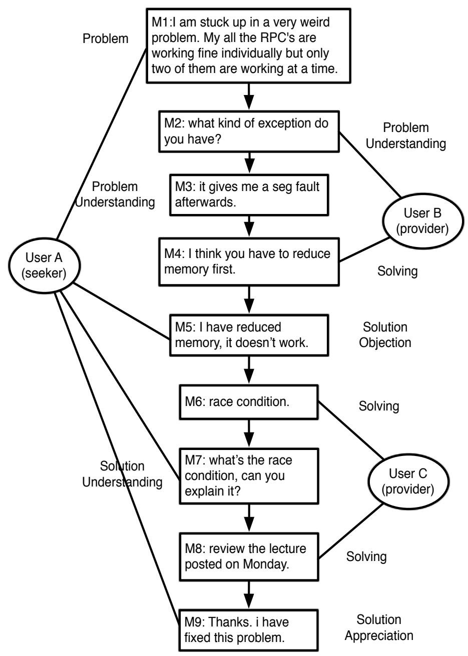


Fig.1. An example of discussion thread.

We use discussion corpus from undergradu-ate operating systems courses. The courses contain programming projects, and students use discussions to share problems and get help from the instructor and other students. Figure 1 shows an example discussion thread with a sequence of message. User A, B and C represents the participants. User A initiates the thread by describing the problem and asks for help. User B asks for more details related to the problem and User A provides some information. User B then gives a possible solution and User A complains that it doesn't solve the problem. User C offers another answer, and User A asks a related question. User C provides an additional suggestion. Finally, User A acknowledges the help with thanking.

Through analyses of the discussion corpus, we identified six distinctive and fre-quent states. User roles are relevant to characterizing the states: information seeker and information provider, and often the role of a user stays the same within a short dis-cussion thread [16]. The first state (Problem or P) is presented by a Seeker. In Figure 1, M1 can be regarded as a P state. In the second state (Problem Understanding or PU), the problem is further elaborated and discussed. PU can consist of multiple messages. Another discussant (student or the instructor) may post a question in order to under-stand the problem that the seeker confronts. Such questions are usually followed by an answer by the seeker who posted the problem. For example, M2 and M3 help the participants understand the problem. In the third state (Solving or S), a participant provides a direct solution or a hint. Although we label it as S, the grammatical form for such messages may vary. For example, hints can be provided as a question:"why not try ABC?". After Solving, the seeker (or other participants) can respond with Solution Appreciation (SA), Solution Objection (SO) or Solution Understanding (SU). In SA, seeker can acknowledge the assistance with thanking, like M9.

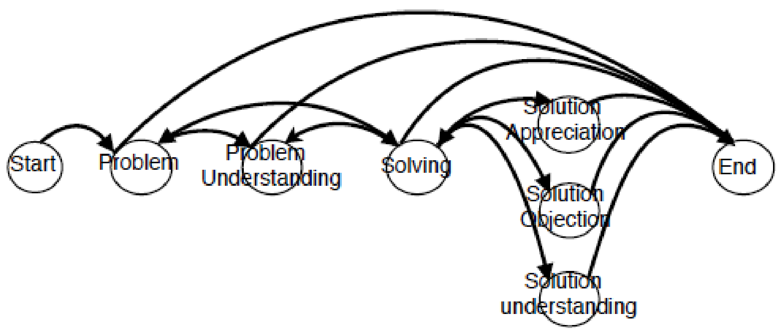


Fig. 2. State transition model for Q&A discussions

Table 1. A Q&A State Model: Definitions and Examples.

State	Definition	Example	Count	Kappa
Problem (P)	Original problem is proposed by information seeker	I stuck in a very weird problem…..	251	0.98
Problem Understanding (PU)	1.Providers ask related questions for understanding original question 2. Seekers answer the related questions and supply more details related to original issues.	1.What kind of exception do you have? 2. It's seg fault afterwards	49	0.96
Solving (S)	Information providers supply answer or suggestions for solving original question	You can try to reduce the memory	447	0.99
Solution Appreciation (SA)	Seekers solve the problem and acknowledge the help from providers	It works, Thanks.	25	0.92
Solution Objection (SO)	Seekers find the answer doesn't work and may ask for more help.	It doesn't work, any ideas?	18	0.88
Solution Understanding (SU)	Seekers may be confused about answer and ask questions for understanding.	What's the race condition, can you explain it?	108	0.97

Note that not all of the messages containing the 'thank' words can be labeled as SA because some P messages can contain 'thanks' in advance before a solution is provided. In SO, the seeker or another participant objects or criticizes the answer proposed by a provider, as shown in Figure 1. SU may appear when the seeker fails to understand the solution and may ask for more information. M7 is an example. Note that it is hard to identify the difference between PU and SU only based on the content of the message because similar words may be used in both states. However, the context or the dialogue state of the message can help distinguishing the two. In Figure 2, we illustrate transitions among these states. P state can be followed by a PU as well as a S but its transition to a SA, a SO, or a SU is rare.

Table 1 presents a description of each state and examples. The state information is annotated manually and the last column shows the Kappa values for agreement between two annotators. The table also shows the distribution of the states. We can find that almost 50 percent of states belong to S. There is a small number of SOs.

Table 2. State transition matrix frequencies

state	P	PU	S	SA	SO	SU
P	-	14	220	-	-	-
PU	-	20	19	-	-	-
S	9	16	101	22	17	92
SA	-	-	4	4	-	3
SO	-	-	13	-	-	-
SU	-	-	90	-	-	10

Table 2 shows the frequency of state transitions. We can find that S is a bridge between the first two states and the last three states. The first two states (P and PU) discusses about the problem to be solved, while the last three are the feedback to the solution, and S connect the two parts. S dominates in the corpus. A S often directly follows a P, but there are cases where the Q&A process goes through a PU. Below we examine frequent patterns in the discuss process using the state information.

3. Automatic Discussion State Classification

236 threads and 899 posts are used for constructing the state transition model.

Data preprocessing, normalization, and feature generation

Student discussion data is highly noisy due to variances and informal nature of student written messages. The data pre-processing steps convert some of the informal expres-sions. For example, “yep”, “yeah” and “yea” are all substituted by “yes”. “what’s” and “wats” have to be converted to “what is”. The features for state classification are generated from (a) the message content, (b) neighboring messages, and (c) the mes-sage/author locational information:

-F1: n grams features within current message

-F2: position of the current message, such as the first message, the last message

-F3: position of participants, like the first author, the last author

-F4: n grams features within the previous message

-F5: position of the previous message

-F6: position of previous author

Table 3. Top 3 features for each state

P	PU	S	SA	SO	SU
[get] unigram103_ NotFirst	[get] unigram103_NotF irst	2ndAuthor	[correct] unigram197_Botto m	1stAuthor	[fine] unigram330_ NotFirst
[somehow+delet] bigram65_ Any	[somehow+delet] bigram65_ Any	[get] unigram103_ NotFirst	[Cat _WH+should] bigram421_ Any	replyTo2ndMessage	[it+okai] PRE_bigram248_Bot tom
2ndAuthor	[about] PRE_unigram134 _ Any	[somehow+delet] bigram65_ Any	[Cat _Subj_IWE+had] bigram581_Any	[give+Cat_Objective _IWE] bigram154_ NotFirst	[Cat _BE+wrong] PRE_bigram393_ NotFirst

Given the full features generated from the content and the position, we use In-formation Gain [15] to reduce the features space. We select the 1620 features. The top 3 features for each state are shown in Table 3. Some of the features are n-grams from the current message or the previous message, e.g., a unigram CAT_ISSUE and a bigram not+sure. PRE represents feature from the previous message. Top/Bottom/Any/NotFirst represent position of the cue words in the message.

Linear CRF and other machine learning methods

Linear CRF [9] is a probabilistic model for characterizing the sequential data, referring the feature information, as well as the dependency among neighbors. The probability function are presented in equation (1) as follows,

$$\mathrm{p}\left({}^{Y}/_{X}\right) = \frac{1}{Z(x)} \exp\left\{\sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}, x_t)\right\} \qquad (1)$$

where Z(x) is an instance-specific normalization function, defined as equation (2),

$$Z(X) = \sum_{y} \exp\left\{\sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}, x_t)\right\} \qquad (2)$$

Y is the sequence data, X is the feature vectors with the total number K.   is a parameter vector, and the corresponding feature functions are defined as .

In our application, each thread, containing several messages, is regarded as the sequence data. Linear CRF can capture the dependence among these messages, and give a most likely state transition with the purpose of characterizing each state for each message in a thread. We use Mallet [7] to create the model. Other machine learning methods such as SVM, decision tree, and logistic regression are widely used in practice. Since differences among states are rather clear and the data space is partitional, decision tree can build the model by separating feature space iteratively. SVM is also used as it is sensitive to the data points near the states' boundaries and has been suc-cessfully used for many problems. Logistic regression is another effective algorithm for categorical variables. Weka [10] was employed.

Resampling

We apply sampling methods due to unbalanced data. We split the six states as majority classes, including P, S and minority classes containing the rest four states. Because SVM, decision tree and logistic regression regard each message independently, resampling method can be applied directly by adding a copy of each minority instance.

As linear CRF rely on the message sequence, we separate threads as majority and minority classes. Majority thread can be defined as threads that have only P and/or S state, while minority threads include at least one message with other states: PU, SA, SO and SU. A combination of downsampling and upsamping methods is utilized for balancing the data and obtaining the better results; we reduce the majority threads by 30% and duplicate minority threads twice. For each classifier, we performed 10-fold cross-validation. In each fold, we separate data randomly, and use 80% for training data and 20% for test. Resampling is done for training data only.

<u>Classification Results</u>

Table 4. Classification Results

	Precision/Recall/F-measure (%)					
Model	**P**	**PU**	**S**	**SA**	**SO**	**SU**
Linear CRF	98.1/95.3/**96.7**	32.0/20.6/**25.0**	86.4/90.6/88.5	43.1/38.8/40.8	23.3/12.4/16.2	62.2/74.0/**67.5**
SVM	100/93.8/**96.7**	15.8/36.7/22.1	88.7/91.1/**90.0**	42.1/63.0/**53.6**	24.1/56.7/**31.2**	53.8/90.6/**67.5**
J48	99.6/94.1	10.1/28.7/15.8	83.0/89.0/85.2	22.5/48.8/29.1	10.8/23.3/14.3	47.6/80.1/59.5
LR	87.2/87.5/87.3	12.1/22.7/15.8	85.8/87.9/85.2	41.0/56.3/29.1	22.8/15.0/14.3	41.8/59.6/59.5

Table 4 shows precision, recall and F-measure scores for different classifiers. Linear CRF, SVM perform better than logistic regression and decision tree. It seems that the relation between states and features are not fully captured through a non-linear function directly. Although SVM and decision tree regard messages individually, both methods make use of dependencies among neighboring messages as some of the features capture previous message content and location information. Because of the small size for state PU, SA and SO, the precision and recall for these three states is low, especially for decision tree, which is sensitive for the features and instances. The precision and recall for state SA is relatively high. A possible reason is that its features include useful cue words including “thanks”, ”it works” that appear regularly. On the other hand, although we have 108 instances for state SU, the precision and recall for it is not so high. We may need further examples due to its variances. Another reason is that SU often contains a question for the solution, which may use similar key words as in P, thus it’s challenging to completely distinguish SU from P.

4. Analyzing Q&A Process with State Information

<u>Frequent dialogue patterns</u>

We use the classified information in analyzing frequent state transitions and dialogue patterns. State transitions are represented as a sequence of three states: “ previous state -> current state -> next state”. We further distinguish contributions by the instructor and students. The end of discussion is labeled as “end”. We list the top ten frequent patterns from 236 discussion threads in Table 5.

Table 5. The top ten frequent patterns for both instructor and students

Instructor			Student		
pattern	count	percent	pattern	count	percent
P->S->end	88	13.31%	S->SU->S	77	11.65%
P->S->SU	36	5.45%	P->S->S	33	4.99%
SU->S->end	30	4.54%	S->S->S	26	3.93%
S->S->end	20	3.03%	P->S->end	24	3.63%
SU->S->SU	17	2.57%	SU->S->S	16	2.42%
S->S->SU	12	1.82%	S->SO->S	13	1.97%
P->S->PU	8	1.21%	S->S->end	13	1.97%
PU->S->end	8	1.21%	S->S->SU	12	1.82%
S->S->S	7	1.06%	S->SU->SU	9	1.36%
SU->S->S	6	0.91%	SU->SU->S	9	1.36%

The trends include:

a. Most SUs are generated by students.

b. The most frequent pattern for instructors is "P->S->end", and its frequency is much higher than the corresponding students' pattern. It indicates that instructor's answers can end many discussions, and may discourage further participation by the student.

c. If the previous state is P, most of the current states, generated either by the instructor or the students, is S. Instructor answers may be followed by SU: "P->S->SU", In contrast, students' S states tend to be followed by another S. That is, additional or different answers are proposed to students' answers more often than to instructors' answers.

d. If the previous state is SU, the instructor tends to post S, and the next state is often SU. Given a SU, students post either S or SU, and it can follow by another S.

e. If the previous state is S, students tend to post US, which is followed by a S. This is the most frequent pattern for students. Students can also post a S in response to S, which can be followed by another S. The second most frequent pattern is "S->S->S".

f. If the previous state is PU, the instructor tends to post a S. Students may post PU in response to a PU, which is followed by S or PU. Generally speaking, students may need more discussion turns to comprehend the problem.

<u>Timing of responses</u>

Table 7 lists frequent state transitions based on time information. "N/A" means that there is no such state transition in the instructor pattern. 'Instructor' columns represent time interval values when the current message is posted by the instructor. Likewise, 'Student' columns show time intervals when the current message is posted by a student. According to the Table 7, we can observe the following.

Table 7. Time interval for state transition

Previous state ->Current state	Instructor		Student	
	Median	Mean	Median	Mean
P->S	4:38:39	7:56:37	1:55:11	5:29:28
P -> PU	3:36:7	6:23:6	3:9:58	3:37:16
PU -> PU	1:37:32	1:4:21	2:16:4	8:19:21
PU -> S	4:27:53	8:16:52	0:57:45	5:44:10
S -> S	5:25:58	8:49:43	1:34:26	5:41:39
SU -> S	4:22:19	8:10:59	1:18:58	3:22:22
S -> SA	1:4:37	4:30:39	0:45:54	2:17:21
S -> SO	N/A	N/A	1:55:21	5:2:18
S -> SU	N/A	N/A	1:59:2	9:1:9

1. From P state to S state, usually students spend less time in posting S than the instructor.

2. Student will spend less time to positively acknowledge (correct) answers. In other words, SA is quickly followed by a S. Transitions from S to SA, SO, and SU take a longer time. If the answer doesn't work, students may spend more time to check their problem and work.

3. The most time consuming state transition is when the instructor posts S in response to a S.

4. Usually, students reply messages more promptly than the instructor.

5. Resolved/Unresolved Discussion Classification

A discussion thread is 'resolved' when all the problems proposed by the participants, including initial problems, derived problems are solved successfully. Otherwise, it's unresolved thread. The features used for thread classification are:

-F1: n gram features within the final message in a thread

-F2: position of the final message, such as the first message, the last message

-F3: position of the final author, (the first author, the last author)

-F4: n gram features within the previous message

-F5: position of previous message and previous poster

-F6: state information

Table 8 presents the thread classification result. Comparing the three tables, we can conclude that state information indeed improve the performance of classifiers for thread classification. The state information represents the role of the message and effectively abstract low-level feature content or locational features. The state infor-mation also captures the dependencies among the messages within the whole thread, and can provide additional context information. For example, if the last state is PU, without state information, the message can be labeled as S for the understanding problems, and the classifier may label it as the resolved because it provides a solution. Generally, such abstractions provide better performance in machine classification when training data is not enough [15]. They also assist human analysis. The thread classification can help instructors in distinguishing resolved vs. unresolved discussions. Furthermore, state information helps instructors have insight on the process of discus-sion and facilitate them to understand the current state of the discussion. Such infor-mation supplies suggestions for instructors to decide when or whether to participate in the discussion.

Table 8. Precision, Recall and F-measure for thread classification

(a) Without state information

	Resolved			**Unresolved**		
	Precision	**Recall**	**F-value**	**Precision**	**Recall**	**F-value**
J48	0.92	0.94	**0.93**	0.71	0.66	**0.68**
SVM	0.87	0.98	0.92	0.81	0.39	0.52
LR	0.90	0.90	0.90	0.55	0.55	0.55

(b) With annotated state information

	Resolved			**Unresolved**		
	Precision	**Recall**	**F-value**	**Precision**	**Recall**	**F-value**
J48	0.95	0.99	**0.97**	0.94	0.75	**0.84**
SVM	0.90	0.98	0.94	0.85	0.50	0.63
LR	0.92	0.93	0.93	0.68	0.64	0.66

(c) With classified state information

	Resolved			Unresolved		
	Precision	Recall	F-value	Precision	Recall	F-value
J48	0.93	0.97	**0.95**	0.88	0.71	**0.78**
SVM	0.88	0.97	0.93	0.84	0.51	0.64
LR	0.91	0.90	0.90	0.64	0.66	0.65

5. Related work

There has been prior work on discussion analysis including use of speech act framework in modeling online discussions [3,4,5]. Some people focus on the roles that students play such as asking problems or answering other's questions [12,13]. Hidden Markov Model provides the framework for modeling the dialogue structure with hidden states [1,2,11]. They are closely related to our work, and we extend the existing framework by closely modeling the dialogue development and information exchange in Q&A discussions. In particular, we explicitly model problem and solution understanding phases as well as question and answer phases, and analyze the information exchange process using the state information. Graph-based approaches have been used in text mining, clustering and other related problems including labeling dialogue with tutors [1]. In order to facilitate the analysis of student discussions, we extend the existing work and represent a discussion thread as a graph model where each state in the model represents a message. There has also been work on machine classification of student online discussions [8,12,14] and results have been used to find meaningful dialogue patterns including features for critical thinking. Our work complements these results by closely examining and classifying Q&A processes.

6. Conclusion

We have presented a graph model for analyzing the discussion process and developed approaches for message state classification and thread characterization. The state information is used in analyzing frequent patterns and time intervals, and identifying different roles that instructors and students play in the Q&A process. Thread classifi-cation for resolved vs. unresolved problem is supported by the state information. As a next step, we plan to collect more data in order to obtain the more reliable classification result and explore additional improvement, including topic-based analysis of student problems. We plan to evaluate usefulness of the information with instructors.

References

[1] Boyer, K. E., Ha, E. Y., Wallis, M. D., Phillips, R., Vouk, M. A., & Lester, J. C. (2009). Discovering tutorial dialogue strategies with hidden Markov models. Proc. AIED 2009.

[2] Chi, M., VanLehn, K., & Litman, D. (2011). An Evaluation of Pedagogical Tutorial Tactics for a Natural Language Tutoring System: A Reinforcement Learning Approach. IJAIED.

[3] Forbes-Riley, K., & Litman, D. (2009). Adapting to student uncertainty improves tutoring dialogues. Proceedings of International Conference on Artificial Intelligence and Education.

[4] Fossati, D., Di Eugenio, B., Ohlsson, S., Brown, C., & Chen, L. (2010). Generating proactive feedback to help students stay on track. Proc. of ITS Conference 2010.

[5] Hirschberg, J. and Litman, D. (1993). Empirical Studies on the Disambiguation of Cue Phrases, Computational Linguistics, 19 (3).

[6] Kim, J., et al., (2006). Mining and assessing discussions on the web through speech act analysis. Workshop on Web Content Mining with Human Language Technologies at ISWC.

[7] McCallum, A. (2002). MALLET: a machine learning for language toolkit. http://mallet.cs.umass.edu

[8] McLaren, B. et al., Using Machine Learning Techniques to Analyze and Support Mediation of Student E-Discussions, Proceedings of AIED 2007.

[9] Sutton, C. (2006). An Introduction to Conditional Random Fields for Relational Learning.

[10] Witten, I.H., Frank, E.(2005). Data Mining: Practical machine learning tools and techniques. 2nd edn.

[11] Seo, S., Kang, J., Drummond, J. and Kim, J. (2011). Using Graphical Models to Classify Dialogue Transition in Online Q&A Discussions , Proceedings of AIED 2011.

[12] Mu, J., Stegmann, K., Mayfield, E., Rose, C., and Fischer, F. (2012). The ACODEA framework: Developing Segmentation and Classification Schemes for Fully Automatic Analysis of Online Discussions. Proc. CSCL 2012.

[13] Ravi, S. and Kim, J. (2007). Profiling Student Interactions in Threaded Discussions with Speech Act Classifiers. Proceeding of AIED 2007.

[14] Rus, V., Graesser, A., Moldovan, C., and Niraula, N. (2012). Automatic Discovery of Speech Act Categories in Educational Games, Proceedings of EDM.

[15] Saeys, Y., Inza, I., and Larranaga. P. (2007). A Review of Feature Selection Techniques in Bioinformatics. Bioinformatics, 23(19):2507-2517.

[16] Yoo, J. and Kim, J. (2012). Predicting Learners Project Performance with Dialogue Features in Online Q&A Discussions. Proceedings of ITS 2012.

Extending Collaborative Learning Modeling with Emotional Information

Olga C. Santos[1], Jesus G. Boticario[1], Raúl Cabestrero[2], Pilar Quirós[2]

[1] aDeNu Research Group. Artificial Intelligence Dept. Computer Science School. UNED
C/Juan del Rosal, 16. Madrid 28040. Spain
[2] Basic Psychology Dept. Faculty of Psychology. UNED.
C/Juan del Rosal, 10. Madrid 28040. Spain
[1]{ocsantos,jgb}@dia.uned.es, [2]{rcabestrero,pquiros}@psi.uned.es

Abstract. This work presents some initial ideas on a data mining based approach for building affective collaborative systems. In particular, we focused on the modeling issues involved in providing open affective student interaction models by using data mining techniques. The approach facilitates transferability and analysis without human intervention, and extends with emotional information previous data mining based developments.

Keywords: Collaboration, Data mining, Open models, Affective Computing.

1 Introduction

Given that affective issues play a significant role in e-learning scenarios [1, 2], in the context of the MAMIPEC project we are investigating emotions modeling in Computer Supported Collaborative Learning (CSCL), where either positive or negative emotions can emerge [3]. Positive ones are expected to bring about an increase in the number of users' interactions and accordingly the development of new collective generated knowledge. On the other hand, when individual aims collide with collective ones, negative emotions frequently arise. Under CSCL learners usually cope with more striking challenges than those present under face-to-face learning [4]. For instance, objectives of some group members can interfere with ones of others. Also, diversity in terms of levels of involvement, working styles and interaction modes frequently become overlapped within the group members. Additionally, the lack of previous common background and generally accepted point of view usually obstructs the way of getting cooperative solutions [3].

In this context, provided that Data Mining (DM) can be used for emotional information detection in CSCL [5], our goal is to extend the Collaborative Logical Framework (CLF) collaboration model [6] with emotional indicators and personality traits following a DM approach used in previous collaboration experiences [7].

2 Affective Collaborative Modeling approach

Personality traits and emotions play a key role in social and collaborative scenarios [4]. In this sense, it can be stated that personality can modulate the way the student participates on a given situation. For instance, some studies have found that participants that exhibit lower scores on extraversion and higher on mental openness prefer on-line learning tend [8]. Thus, in order to enrich adaptation in collaborative learning scenarios with affective support, the model has to take into account the user personality traits that can be influencing the user interaction behavior. It has also to consider the user affective state (i.e. pride, shame, curiosity, frustration, etc.) generated within the undergoing activity itself and the whole CSCL interaction. For this, i) context, ii) process and iii) assessment are considered key issues to model collaboration [9, 7].

The ***collaboration context*** affects students' potential and their capacity to collaborate. Information comes from data related to both students and the environment, which should be relevant to students' teamwork skills [10]. This information can be collected in the collaborative learning experience from an initial questionnaire (e.g., personal, academic and work-related data, study preferences, and personality traits).

The ***collaboration process*** involves features such as activity, initiative or acknowledgment. Relevant information can be obtained by analyzing students' interactions in communication tools such as forums [11] because of the close relationship that exists between students' collaboration and interactions. In this sense, previously we proposed a statistical analysis of the interactions in forums to discover some features that make students suitable for collaboration [6], namely student initiative, activity and regularity, as well as perceived reputation by their peers. Students' regularity indicators involve time variables because the interactions are considered over a period of time. In any case, these metrics are general in as much as they are based on non-semantic statistical indicators (e.g. number of replies, regularity of interventions, etc.) and thereby flexible enough to be potentially instantiated in diverse collaborative environments. In order to take into account affective information in these collaboration indicators, several information sources such as physiological data, keyboard and mouse interactions, explicit subjective affective information provided by learners, facial expression, etc. gathered while learners collaborate in the environment can be considered [12].

To cover aforementioned key issues, the approach we have been following offers ***collaborative assessment*** metrics based on DM process (clustering) to facilitate transferability and analysis without human intervention [7]. It also follows the open model strategy, which has shown its benefits in the educational context. This strategy uses scrutable tools that enable students to access inferred models and actively intervene in the modeling process [13], this way raising metacognitive information [7].

Our proposal for affective collaborative learning modeling is depicted in Fig. 1. In particular, to account for affective issues in the given *collaboration context* (user and environment), the approach has to be extended with an analysis of the affective reactions, elicited during the *collaboration process* within the ongoing collaboration *task* itself, and those due to the *interaction with peers* that feed the *collaboration assessment* and produce not only the *statistical indicators* proposed in [6] but also the add *affective ones*. The *affective indicators* are to be calculated with DM techniques in the light of the *collaboration assessment* by means of the *interaction content*: positive (proposing or suggesting; supporting or agreeing), negative (opposing or disagreeing) or ambivalent (information giving; inquiring; answering or specifying) as rated by both the emitter and the receiver (*interaction ratings* using weather 'overt' – subjective reports– or 'covert' –physiological or behavioral recordings– sources of information) [14]. To cope with *interactions latency*, it has to be taken into account if interaction are produced within certain time window or never take place at all –e.g. unanswered message–. On top of that, the *roles* could elicit an additional emotional reaction or modulate existing ones. Two different types have to be considered: *scripted* and *naturally emerged*. First ones are externally assigned, as a consequence of the statistical interaction indicators (i.e. information gatherer, moderator in the CLF *task* [6], etc.). Second ones emerge naturally in any collaborative work situations (i.e. task or social leadership or other types of roles that emerge in learning situations).

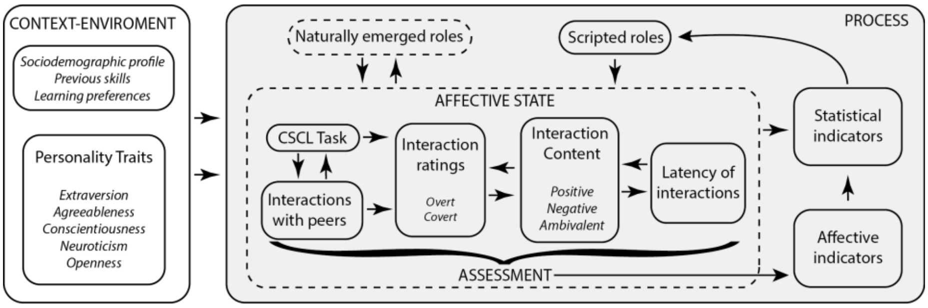


Fig. 1. Affective enriched statistical indicators in the open affective learner model

3 On-going work

To investigate how to enrich the statistical indicators with the affective ones, a CLF collaborative task was set out in Madrid's Week of Science 2012 with a total participation of 17 participants (including pilot experiments). They were asked to collaboratively solve one conundrum on a given time frame following three consecutive stages (*individual*: each participant proposes solution; *collaboration*: discussions and ratings among participants to enrich individual solutions; and *agreement*: solution proposed by moderator and discussed and rated by the rest of participants) while their collaboration interactions and affective information (i.e. personality questionnaires, physiological and behavioral recordings and subjective reports) are processed [6].

All these sources of information, along with the statistical indicators, deserve future analyses in order to refine and calibrate affective indicators and to articulate them using a DM approach. By introducing aforementioned affective issues the approach is expected to improve collaborative learning. In particular, based on our experience in developing educational recommender systems [15] those affective indicators detected will serve to develop affective educational recommendations.

Acknowledgments

Authors would like to thank the Spanish Ministry of Economy and Competence for funding the MAMIPEC project (TIN2011-29221-C03-01).

References

1. Kreijns, K., Kirschner, P.A., Jochems, W.: Identifying the pitfalls for social interaction in computer-supported collaborative learning environments: a review of the research. Computers in human behavior 19, 335-353 (2003)
2. O'Regan, K.: Emotion and e-learning. Journal of Asynchronous learning networks 7, 78-92 (2003)
3. Järvenoja, H., Järvelä, S.: Emotion control in collaborative learning situations: Do students regulate emotions evoked by social challenges? British Journal of Educational Psychology 79, 463-481 (2009)
4. Solimeno, A., Mebane, M.E., Tomai, M., Francescato, D.: The Influence of Students and Teachers Characteristics on the Efficacy of Face-to-Face and Computer Supported Collaborative Learning. Computers & Education 51, 109-128 (2008)
5. Calvo, R.A.: Incorporating Affect into Educational Design Patterns and Frameworks. Ninth IEEE International Conference on Advanced Learning Technologies, 2009 (ICALT 2009), 377-381 (2009)
6. Santos, O.C., Rodríguez, A., Gaudioso, E., Boticario, J.G.: Helping the tutor to manage a collaborative task in a web-based learning environment. In: AIED2003 Supplementary Proceedings, 153-162. (2003)
7. Anaya, A.R., Boticario, J.G.: Content-free collaborative learning modeling using data mining. User Modeling and User-Adapted Interaction 21, 181-216 (2011)
8. Santo, S.A.: Virtual learning, personality, and learning styles. vol. 62. ProQuest Information & Learning, US (2001)
9. Topping, K.J.: Methodological quandaries in studying process and outcomes in peer assessment. Learning and Instruction 20, 339-343 (2010)
10. van Gennip, N.A., Segers, M.S., Tillema, H.H.: Peer assessment as a collaborative learning activity: The role of interpersonal variables and conceptions. Learning and Instruction 20, 280-290 (2010)
11. Perera, D., Kay, J., Yacef, K., Koprinska, I.: Mining learners' traces from an online collaboration tool. Workshop Educational Data Mining, 13th International Conference of Artificial Intelligence in Education, 60–69 (2007)
12. Santos, O.C., Salmeron-Majadas, S., Boticario, J.G.: Emotions detection from math exercises by combining several data sources. LNAI, 7926, 742–745 (2013)
13. Bull, S., Gardner, P., Ahmad, N., Ting, J., Clarke, B.: Use and trust of simple independent open learner models to support learning within and across courses. In: Houben, G.-J., McCalla, G., Pianesi, F., Zancanari, M. (eds.) User Modeling, Adaptation, and Personalization, pp. 42-53. Springer (2009)
14. Nummenmaa, M.: Emotions in a web-based learning environment. Annales Universitatis Turkuensis, B 304 (2007)
15. Santos, O.C., Boticario, J.G., Manjarrés-Riesco, A.: An approach for an Affective Educational Recommendation Model. Recommender Systems for Technology Enhanced Learning: Research Trends & Applications. Manouselis, N., Drachsler, H., Verbert, K., Santos, O.C. (eds.). Springer, 2013 (in press).

Analysis of Emotion and Engagement in a STEM Alternate Reality Game

Yu-Han Chang, Rajiv Maheswaran, Jihie Kim, and Linwei Zhu

University of Southern California
Information Sciences Institute
Marina del Rey, CA 90292
{ychang,maheswar,jihie}@isi.edu, vic90228@gmail.com

Abstract. Alternate reality games (ARGs) are a promising new approach for increasing student engagement; however, automated methods for analyzing and optimizing game play are non-existent. We captured the player communication generated by a recent STEM-focused ARG that we piloted in a Los Angeles charter high school. We used shallow sentiment analysis to gauge the levels of various emotions experienced by the players during the course of the game. Pre/post-game surveys gauged whether the game narratives had any effect on student engagement and interest in STEM topics.

1 Introduction

Alternate Reality Games (ARGs) are a relatively new genre that has shown promise for engaging students in STEM learning activities. These transmedia experiences typically draw participants into fictional narratives, where players interact via various forms of social and traditional media, and frequently become part of the storyline themselves. They differ from traditional virtual reality computer games, where the entire story takes place in a fictional online world. In ARGs, the game world overlaps with the real world. Players visit real places, research the real world wide web, communicate with other players and fictional characters using real social media, phone, text messaging, and occasionally live encounters in the real world. For education, this novel game format has the potential to literally bring science activities and learning into the normal lives of students, emphasizing STEM relevance to the students context, surroundings, and community. The ARG brings the game space into the physical daily reality of students [**?**,**?**].

In this paper, we describe a pilot ARG we designed and implemented at USC Hybrid High in Fall 2012. We describe the ways in which we were able to capture player data, both by observing the players in game, and by validating these observations through pre and post game tests. In order for ARGs to truly support educational objectives, we need to be able to unobtrusively measure and understand the performance of players within the game, using only their in-game, visible interactions, such as website visitation and forum postings. Individual player assessment enables puppetmasters to tweak the game play to maximize engagement and educational outcome for each learner. Clearly AI and other computational techniques are needed to reach this goal, and this short paper only presents a summary of a small step in this direction.

Fig. 1. (Top Left) The main characters in the game: William, Isa, and Rudy, (T. Right) The final story element in the game, where Fortinbras' CEO is arrested. (Bottom Left) Special trip to Space X facilities, (B. Center) Mysterious poster at USC Hybrid High, (B. Right) Device used to thwart Fortinbras.

USC Hybrid High ARG Pilot: Operation Daylight. In Fall 2012, we fielded a pilot alternate reality game, "Operation Daylight," at USC Hybrid High, a new charter high school with approximately 100 ninth graders in its inaugural class. The population is almost entirely minority and receive free/reduced lunches. The game focuses on π, an organization set up centuries ago to defend science. Its most recent incarnation, i4, needs students from USC Hybrid High to be their next generation, and the game begins with i4 recruiting and training students from the school. In the process, the students complete STEM-related activities to advance up the i4 recruitment ladder.

Gradually, the students uncover an evil plot by Fortinbras Industries that threatens their protagonist recruitment agents, the fictional characters Rudy Vanzant and Isa Figueroa, played by local actors in a variety of video sequences. This requires the students to put their newly learned skills to real use in order to save their friends Rudy and Isa. Figure **??** shows some of the elements used in the game. The game ran for approximately five weeks at USC Hybrid High, from 10/18/12 to 11/21/12. It was a completely optional activity that students could engage in if they chose to, with both online, at-school, and out-of-school elements. Students drove over 27,670 page views to the i4 website and posted 1394 messages to the i4 forum.

2 Methodology and Results

We used well-established scales for measuring student interest in STEM topics developed by OECD's Programme for International Student Assessment (PISA) [**?**]. Pre and post game surveys were developed using these scales, and administered to students at USC Hybrid High one week before the game commenced and one week after the game concluded. The surveys included approximately thirty questions where students would respond "Strongly Agree", "Agree", "Disagree", or "Strongly Disagree." The survey also included questions that established basic demographic information, as well as self-reported aspects of game play. In addition to the survey data, we also collected in-game data such as forum visits, messages posted, videos and pictures posted. We also used the Linguistic Inquiry and Word Count (LIWC) text analysis tool to process the messages [**?**] and detect whether they expressed a positive or negative sentiment, or whether the message contained anxiety, fear, or happiness.

Fifty-nine out of the 94 survey respondents indicated that they had heard of i4 and the Operation Daylight game. Twenty-three of the 29 students who signed up on the Operation Daylight website filled out surveys. Among students who played the game, they overwhelmingly thought the game increased their interest in science (48%) or did not change their already positive interest in science (47%). No one ended up having less interest in science.

These responses are corroborated with the students' answers to the OECD science interest questions. Figure **??** shows how the students' science interest levels changed from the beginning of the game to the end of the game, conditioned how often they visited the i4 forum, and on the average length of their posts on the forum. In these graphs, 0 corresponds to "Strongly disagree" (dislike science), and 3 corresponds to "Strongly agree" (like science). We see that there is a correlation between more visits and higher science interest level, as well as between longer posts and higher interest levels. There also appears to be a correlation between longer posts and a larger amount of increase in science interest.

Figure **??** shows that there is a correlation between forum activity and the major game events, such as the main characters being abducted. This suggests that these ARG story elements might promote the higher science interest levels described above. We also analyze the number of messages that contained certain percentages of message words that indicate positive or negative attitude, anxiety, fear, or sadness. It turned out that there is no clear pattern between the story elements and the production of particular categories of words, contrary to our expectation. For example, the abduction of the main character did not obviously produce more messages of fear or negativity. Generally the proportional levels of positive words stays constant during the game, and the levels of negative words

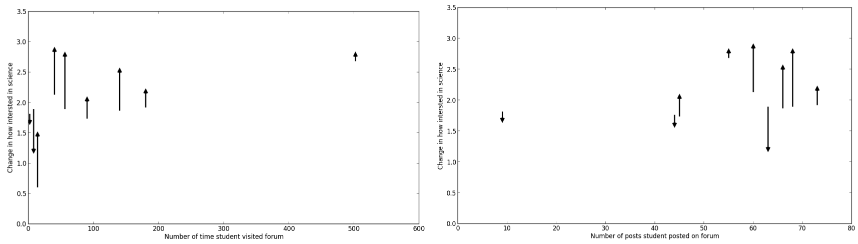


Fig. 2. (Left) Number of visits vs. change in science interest levels, (Right) Average length of forum postings vs. change in science interest levels.

stays quite low. The proportions of messages with varying levels positive words are also shown in Figure **??**. Due to lack of space here, a longer version of this paper will be posted at our website, http://cb.isi.edu.

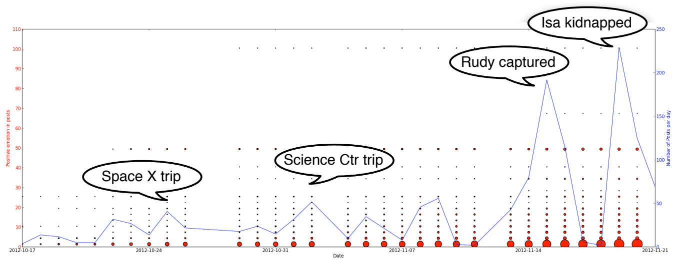


Fig. 3. Time showing the level of forum activity over the course of the game. The thin blue line denotes the number of posts in the forum on each day, the red circles denote how many of those messages contained a particular fraction of positive words.

Acknowledgements. This work was supported by NSF Award #:1224088.

References

1. E. Klopfer. *Augmented learning: Research and design of mobile educational games.* MIT Press, 2008.
2. Herbert W. Marsh, Kit-Tai Hau, Artelt Cordula, Baumert Jurgen, and Jules L. Peschar. Oecd's brief self-report measure of educational psychology's most useful affective constructs: Cross-cultural psychometric comparisons across 25 countries. *International Journal of Testing*, 6(4):311 360, 2006.
3. A. Moseley, N. Whitton, J. Culver, and K. Piatt. Motivation in alternate reality gaming environments and implications for learning. In *3rd European Conference on Games Based Learning*. Academic Conferences Limited, 2009.
4. J.W. Pennebaker. Putting stress into words: Health, linguistic, and therapeutic implications. *Behaviour research and therapy*, 31(6):539–548, 1993.

A Dashboard for Visualizing Deliberative Dialogue in Online Learning[1]

Tom Murray, Leah Wing[1], Beverly Park Woolf

School of Computer Science, [1]Legal Studies Dept.
University of Massachusetts, Amherst, MA

Contact: tmurray@cs.umass.edu

Abstract: New and emerging online trends in group education, work and communication have led to a dramatic increase in the quantity of information and connectivity without always supporting—and sometimes sacrificing—quality. An important opportunity is that online systems can include tools that directly support participants in having higher quality and more skillful engagements. We are evaluating dialogue software features that support participants directly and "dashboard" tools that support third parties (mediators, teachers, facilitators, moderators, etc.) in supporting higher quality deliberation. We will focus on our work in educational settings (college classes) and on our development of a Facilitators Dashboard that visualizes dialogue quality indicators for use as facilitation tools or participant social awareness tools. The Dashboard makes use of text analysis methods to highlight indicators of dialogue quality. We are particularly interested in supporting the "social deliberative skills" that interlocutors need to build mutual understanding and mutual regard in complex or contentious situations.

Keywords: Educational and Knowledge Building dialogue; deliberative skills; scaffolding; multiple representations; dashboards.

1. Introduction

New and emerging online trends in group education, work and communication have led to a dramatic increases in the quantity of information and connectivity without always supporting—and sometimes sacrificing—their quality. An important opportunity is that online systems can include tools that directly support participants in having higher quality and more skillful engagements. We are building and evaluating dialogue software features that support *participants* directly and "dashboard" tools (Few, 2007) that support *third parties* (mediators, teachers, facilitators, moderators, etc.) in supporting higher quality deliberation among participants. In this paper we will focus on our work in educational settings (college classes) and on our development of a Facilitators Dashboard that visualizes dialogue quality indicators for use by

---

[1] A longer version of this short paper appears at www.socialdeliberativeskills.com/papers.

either third parties or participants. We are particularly interested in supporting the "social deliberative skills" that interlocutors need to build mutual understanding and mutual regard in complex or contentious situations (Murray et al., 2013A, B). Prior attempts to facilitate leaner dialogue using visualization and analysis tools, e.g. Asterhan & Swhwatz (2010) and De Groot et al. (2007), tend to focus on argumentation skills, and our work extends or complements this work by focusing on skills more related to mutual understanding and cognitive empathy. Communication, collaboration, and knowledge building have many facets; and we focus our research on a specific area: supporting the social deliberative skills and behaviors that allow interlocutors to build mutual understanding (or "negotiate meaning") in complex or contentious contexts. Recent advances in computational psycholinguistics allow for a more systematic and deeper analysis of dialogues, that is necessary to uncover subtle cues that might be diagnostic of critical deliberation characteristics. In Xu et al. (2013) we report on our work in developing computational methods to measure deliberative skills from online discussions, which have shown promising results. In this paper we will describe our progress and plans for displaying the results of such text analysis in the Dashboard.

2. Dashboard Diagram Pane: Visualizing Key Indicators

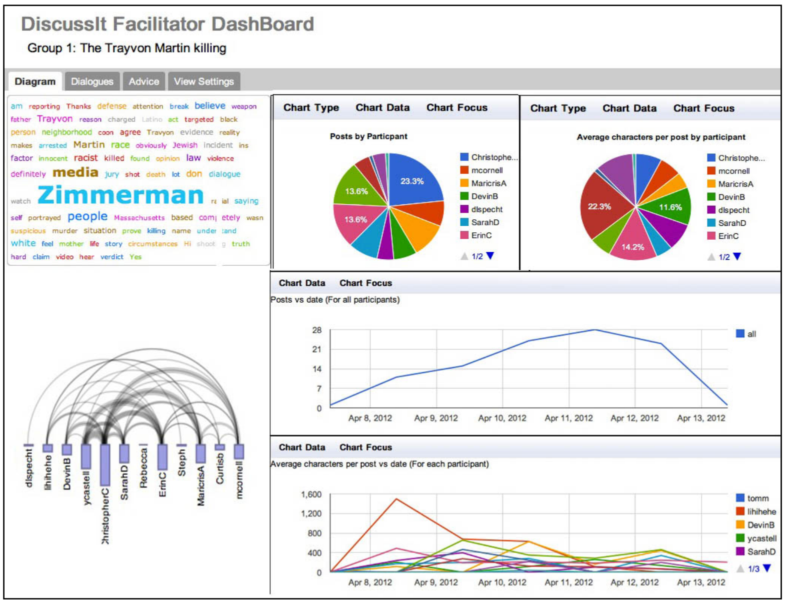


Figure 1: Facilitator Dashboard: Diagram Pane

We have prototyped a Facilitators Dashboard that provides parties a "bird's-eye view" of the state and flow of online engagements. See Figure 1 which shows tools in the "Diagram" tab of the Dashboard. Similar to Iandoli et al., De Groot et al., we visualize user, interaction, and content information, including participation levels, reply networks, and content or theme overviews—in both static and trend (timeline) visualizations. At a more ambitious level, we also use text analysis to identify skillful (or nonskillful) deliberation, emotional tone or sentiment. Further, we have made early forays into automatically identifying dialogue phases (e.g. introductions, deliberation, impasses, persuasion) and turning/infection points or opportunities for intervention (e.g. silences or non-responsiveness, changes of phase or tone, sudden emotional tensions in multiple participants) (Xu et al. 2013).

Figure 2 shows data from a classroom discussion about the fatal shooting of Trayvon Martin by George Zimmerman which was a hot topic in the news during the time of this activity. When the facilitator begins using the Dashboard they select from a list of the deliberation projects, classes, or discussion groups registered with the Mediem software and the Dashboard (not shown in the Figure). Pie and bar charts show participation levels (number of participant posts and average size of posts). Timelines show trends in these same metrics. A social network diagram shows who is replying to whom, with the thickness of the lines proportional to the number of replies. A "word cloud" graphically shows word frequencies through font sizes (the color and location of the words has no meaning in this representation).

3. Dialogue and Advice Panes: Text Analysis

As mentioned above, one component of our project is researching automatic text analysis and machine learning algorithms (and soon also relationship networks) to identify deliberative skill, other indicators related to dialogue quality, and trends or opportunity points (and see Rosé et al. 2008). Text analyses methods have advanced significantly in recent years. According to Graesser et al. (2009) the "increased use of automated text analysis tools can be attributed to landmark advances in such fields as computational linguistics, discourse processes… , cognitive science…, and corpus linguistics…" (p. 34). We are using three types of technologies. The first two, LIWC (Pennebaker et al, 2007) and Cohmetrix (Graesser et al., 2009), are pre-existing text analysis tools that take text segments as inputs and output dozens of measurement or classification metrics. The third technology is a set of machine learning methods we are using that take text, reply and demographic information, and some of the LIWC and Cohmetrix out-

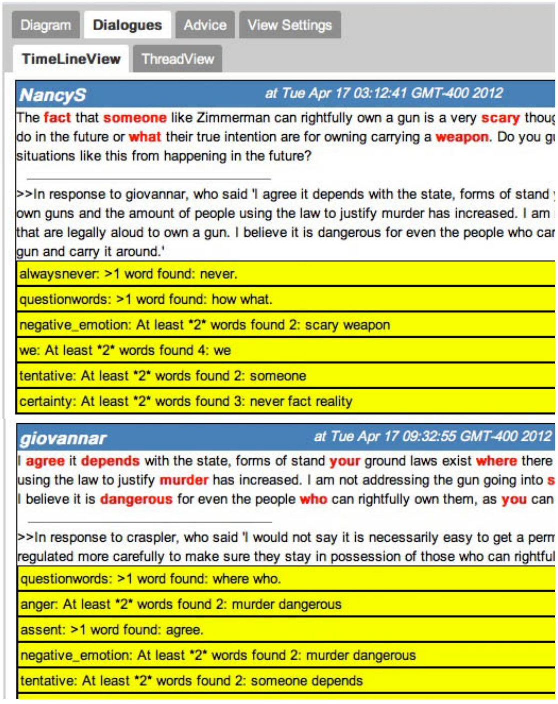


Figure 2: Dashboard: Dialogue Pane

puts as input or training features, and output classification analysis (e.g. whether a segment of text demonstrates good "deliberative skill" or "self reflection").

4. Conclusions

We have described a novel Facilitators Dashboard tool that visualizes dialogue quality indicators for use as facilitation tools or participant social awareness tools that includes textual analysis and described our initial attempts to use it in educational settings. We are particularly interested in supporting the "social deliberative skills" that interlocutors need to build mutual understanding and mutual regard in complex or contentious situations. Developing methods to scaffold SD-skills in online deliberation, for participants and third parties, could have an impact in many online contexts; e.g. knowledge-building, situated learning, civic engagement, and dispute resolution. Students engaged in extended collaborative knowledge building, discussion, or problem solving eventually encounter moments of tension in which they are challenged to understand each other's perspectives and opinions. Engaging with others on complex topics requires not only learning the relevant facts and concepts and making logical inferences but also, engaging with the perspectives and opinions of others who may not share one's views or goals. Doing so requires skills that can be systematically supported. Our work points to how such skills can be supported in online deliberation, collaboration, and dispute resolution—in educational settings and beyond.

5. References

Asterhan, C. S., & Schwarz, B. B. (2010). Assisting the facilitator: Striking a balance between intelligent and human support of computer-mediated discussions. In Workshop on opportunities for intelligent and adaptive behavior in collaborative learning systems (p. 1).

De Groot, R., et al.. (2007). Computer supported moderation of e-discussions: the ARGUNAUT approach. In Proceedings of the 8th international conference on Computer supported collaborative learning (pp. 168-170), July, 2007. International Society of the Learning Sciences.

Few, S. (2007). "Dashboard Confusion Revised." Perceptual Edge, March 2007. From http://www.perceptualedge.com/articles/03-22-07.pdf.

Graesser, A.C., McNamara, D.S., & Louwerse, M.M. (2009). Methods of Automated Text Analysis. Chapter 4 in the Handbook of Reading Research, Volume IV, edited by Michael L. Kamil, P. David Pearson, Elizabeth B. Moje, and Peter Afflerbach. Routledge.

Murray, T., Stephens, A.L., Woolf, B.P., Wing, L., Xu, X., & Shrikant, N. (2013a). Supporting Social Deliberative Skills Online: the Effects of Reflective Scaffolding Tools. Proceedings of HCI International 2013, July, 2013, Las Vegas.

Murray, T., Xu, X. & Woolf, P.B. (2013b). An Exploration of Text Analysis Methods to Identify Social Deliberative Skills. In Proceedings of AIED-13, July, 2013, Memphis, TN.

Pennebaker, J. W., Chung, C. K., Ireland, M., Gonzales, A. L., & Booth, R. J. (2007). The development and psychometric properties of LIWC2007. Austin, TX: www.LIWC.net.

Rosé, C., Wang, Y. C., Cui, Y., Arguello, J., Stegmann, K., Weinberger, A., & Fischer, F. (2008). Analyzing collaborative learning processes automatically: Exploiting the advances of computational linguistics in computer-supported collaborative learning. International journal of computer-supported collaborative learning, 3(3), 237-271.

Xu, X., Murray, T., Smith, D. & Woolf, B.P. (2013) . Mining Social Deliberation in Online Communication: If You Were Me and I Were You . Proceedings of EDM-13, Educational Data Mining, July, 2013, Memphis, TN.

Designing OLMs for Reflection about Group Brainstorming at Interactive Tabletops

Andrew Clayphan, Roberto Martinez-Maldonado, and Judy Kay

School of Information Technologies, The University of Sydney, NSW, 2006, Australia
{andrew.clayphan,judy.kay}@sydney.edu.au
roberto@it.usyd.edu.au

Abstract. Brainstorming is a valuable and widely-used group technique to enhance creativity. Interactive tabletops have the potential to support brainstorming and, by exploiting learners' trace data, they can provide Open Learner Models (OLMs) to support reflection on a brainstorming session. We describe our design of such OLMs to enable an individual to answer core questions: C1) how much did I contribute? C2) at what times was the group or an individual stuck? and C3) where did group members seem to 'spark' off each other? We conducted 24 brainstorming sessions and analysed them to create core brainstorming models underlying the OLMs. We evaluated the OLMs in a think-aloud study designed to see whether learners could interpret the OLMs to answer the core questions. Results indicate the OLMs were effective and that it is valuable, that learners benefit from guidance in their reflection and from drawing on an example of an excellent group's OLM. Our contributions are: i) the first OLMs supporting reflection on brainstorming; ii) models of brainstorming that underlie the OLMs; and iii) a user study demonstrating that learners can use the OLMs to answer the core reflection questions.

Keywords: Open Learner Models, Brainstorming, Reflection

1 Introduction

Brainstorming is a valuable and widely used technique to produce creative solutions to a problem [11]. It is particularly useful when innovation is needed to break out of established ways of thinking, to generate new ideas. When the brainstorming activity is run in small groups, it encourages participants to contribute to the free flow of ideas around a topic, bringing their own creativity, experiences or expertise into play, and increasing the opportunities of enhanced production of rich ideas for the solution. Osborn, the creator [16] promoted the use of brainstorming for creativity. He emphasised that, to be effective, core rules should be followed to reduce members social inhibitions and stimulate idea generation: the focus should be on the *quantity* of ideas; there should be *no early evaluation*; particularly *no criticism*; and *un-usual or divergent ideas welcomed.* Therefore, all participants are encouraged to contribute fully and equally. Discussion should be limited to cases where people are *stuck* and cannot create ideas.

Multi-touch interactive tabletops have proved effective in facilitating face-to-face brainstorming in small-groups [6]. They can support free flow of ideas by providing a shared group interface so that people can generate many ideas in parallel, then interact with digital representations of these ideas, and save the generated ideas offering all team members equal opportunities to contribute [7]. A less explored potential of interactive tabletops is to exploit data about the interaction to capture the processes through the brainstorming session and then show key information about group and individual performance as Open Learner Models (OLMs) [4]. OLMs are those representations of learners' (knowledge, developed skills, performance, understanding, etc.) that are accessible to the learner or group of learners they represent. They can then serve several roles, including support for reflection [5], formative assessment [2] and facilitate collaborative interaction [3]. We particularly focus on the potential value of Open Learner Models (OLMs) as a driver for individuals to reflect on their individual and group performance after a brainstorming session.

The rest of the paper is organised as follows. Next, we outline related research work on OLMs for group work and interactive tabletops. Section 3 describes ScriptStorm, our tabletop system for brainstorming. Section 4 describes the design of our OLM and our evaluation is presented in Section 5. We conclude with a discussion of the results and future work.

2 Related Work

OLMs have been used to facilitate group interaction by enabling learners to identify peers for collaboration [2]. It has been shown that there is value in providing multiple OLM representations helping support higher levels of reflection, because different learners prefer different forms of OLMs, particularly to meet differing concerns [12]. There has also been some exploration of how an ITS can help a learner in brainstorming [18]. Some of the ways such systems can be beneficial is to help learners realise whether they followed recommended practices for brainstorming effectively, particularly in terms of avoiding early evaluation and whether group members suffered blocks [9] in the session.

Some research has started to explore OLM visualisations that represent collaborative learning at interactive tabletops. Martinez-Maldonado et al. [14] validated a set of such OLMs with teachers, showing they could identify the level of collaboration. Al-Qaraghuli et al. [1] presented a visualisation that showed detailed information of students actions at a tabletop over time to foster deep analysis of the process they followed. These authors also provided a small pie chart on the interactive tabletop showing students a real time indication of each learners' participation. Martinez-Maldonado et al. [15] built a dashboard OLM for the teacher to see real-time information about aspects of collaboration for multiple groups in a classroom of interactive tabletops. These examples aimed either to show 'learner models' to the teacher or have been used for research purposes only. Our work goes beyond this by evaluating OLMs that can be presented to learners at an interactive tabletop to promote self-reflection at the end of a brainstorming session. In this sense, it is similar to Do-Lenh's [10] work, for a multi-tabletop classroom where a simple form OLM gave indication of the progress of each group on a wall display for all students to see.

3 Foundations for design of the Open Learner Models

The need for OLMs to support reflection at a tabletop for brainstorming was identified when we evaluated *Scriptstorm* [8], a scripted tabletop brainstorming system (Figure 1). ScriptStorm had three main stages: (1) idea generation − the "storming" to create ideas; (2) idea categorisation − to organise ideas under category headings; and (3) reflection − to support learners by reflection-on-action [17]. While the scripting proved valuable, the reflection stage did not enable participants to appreciate how well they had followed the recommended brainstorming process. We analysed the data from the study to explore how to create OLMs that could provide more effective support for reflection.

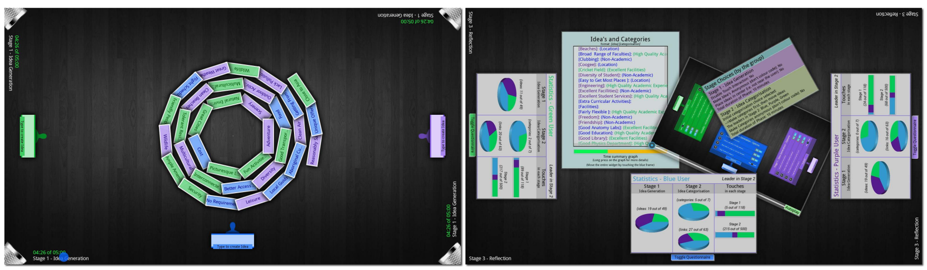


Fig. 1. ScriptStorm: Idea Generation Stage (left), Reflection Stage (right).

We describe *Scriptstorm*, the study, the data collected and the analyses conducted for this work. *Scriptstorm* uses physical keyboards at a multi-touch tabletop. Figure 1-left shows an example table-shot after a group has created several ideas, visible in a circle at the centre of the table. This layout reduces the sense of ownership of ideas and the circular orientation avoids favouring any one user's reading. Ideas are colour coded to indicate the author, giving an indication of each person's level of contribution. Figure 1-right shows the elements available in the reflection stage. Each user has a set of charts showing each person's contributions. Pie charts show how many ideas each person made in Stage 1, how many categories and classification of ideas into them in Stage 2. A bar chart shows touches by each user in each stage. There is a list of the ideas with their categories in the middle, details of the scripting choices made and a replay of the table. Touches were logged by the tabletop and linked to the user making use of a depth camera [13].

The evaluation had 12 groups, each with 3 people (36 participants, 22 male, 14 female, all university students, from diverse degrees – medicine, social science and computer science, aged 19-30, mean age 23). Each group did 2 brainstorms, counter-balanced on scripting condition and topic. Each group was instructed of the rules of brainstorming to follow. Careful analysis of the data indicated the topic and scripting conditions were comparable, making for 24 sessions of data for analysis. All sessions were video recorded.

We analysed the study data to create a model of brainstorming as a foundation for the OLMs. This model provides a bound on the time-between-ideas when the brainstorm is running well. This is important since we can then use it to automatically determine when a group or individual is stuck, and determine if ideas from different users are sparked off each other. Groups created 16 to 104 ideas per session (average = 48; standard deviation = 24), average time between ideas 7.32 seconds (SD = 4.2) range of $2.88 - 17.93$ seconds. We explored the frequency distribution of times, a single hump, slightly left of the peak at 7 seconds. For the individual, average time between ideas was 26.16 seconds (SD = 21.64), range $5.75 - 110.5$ seconds. We arrived at a maximum idle time for a group before being classified as stuck as 22 seconds (mean group time difference + SD), and for an individual 49 seconds (three times the mean). We also used 22 seconds to scan for ideas that potentially sparked other ideas. These values are used as measures in our OLM to highlight interesting periods. Additionally we analysed output in terms of 15 second periods, resulting in a range of 0 to 13 ideas, accounting for outliers, the average being 4 ideas. We used this in our OLM as the basis for a colour coding scheme (red, orange, green), representing: below, average and above average performance.

4 Open Learner Model Design

We needed to enable learners to answer our core questions: *C1) how much did I contribute? C2) at what times was the group or an individual stuck?* and *C3) where did group members seem to 'spark' off each other?* To help learners find answers to these questions, we designed the OLMs in Figure 2 to present six different views of the user trace data. The pie chart (chart 1) shows the number of ideas each person created (C1). Following, there are four aligned timelines. Chart 2 shows when each idea was created with by a dot, the colour of which indicating authorship. The vertical axis indicates the category from the second phase of the brainstorm. Stuck periods are shown as coloured rectangles for the group (2a) and coloured bars for individuals (2b). In the figure the group got stuck twice between 183-209 and 222-244 seconds, the green user stuck between 148-209 and 211-266 seconds, the purple user stuck between 146-245 seconds and the blue user not stuck at all. To model where people sparked off each other, we identified cases where one persons idea was closely followed by another according to the category classification. This measure is shown with yellow bars (2c). There are seven of these in the diagram, for example on category reference 6 between 65-81 seconds (ideas 65s-C, 77s-B, 81s-B). This measure is clearly an inexact measure that is sensitive to the particular categories chosen, however it is indicative of sparking and showing it in an OLM helps users consider this aspect (C2,3). The next timeline (chart 3) shows the performance of each learner in 30 second snapshots (C1,2). The timeline after that (chart 4) shows cumulative progress with segments colour coded according to the rate of contribution (C2). The final timeline (chart 5) is a spectrogram indicating when a group was talking. Learners were instructed to call out each idea they generated in the idea generation stage and we expected discussion if a group was stuck (C2,3). The last view (label 6) is a table with categories and associated ideas annotated with author and time of creation.

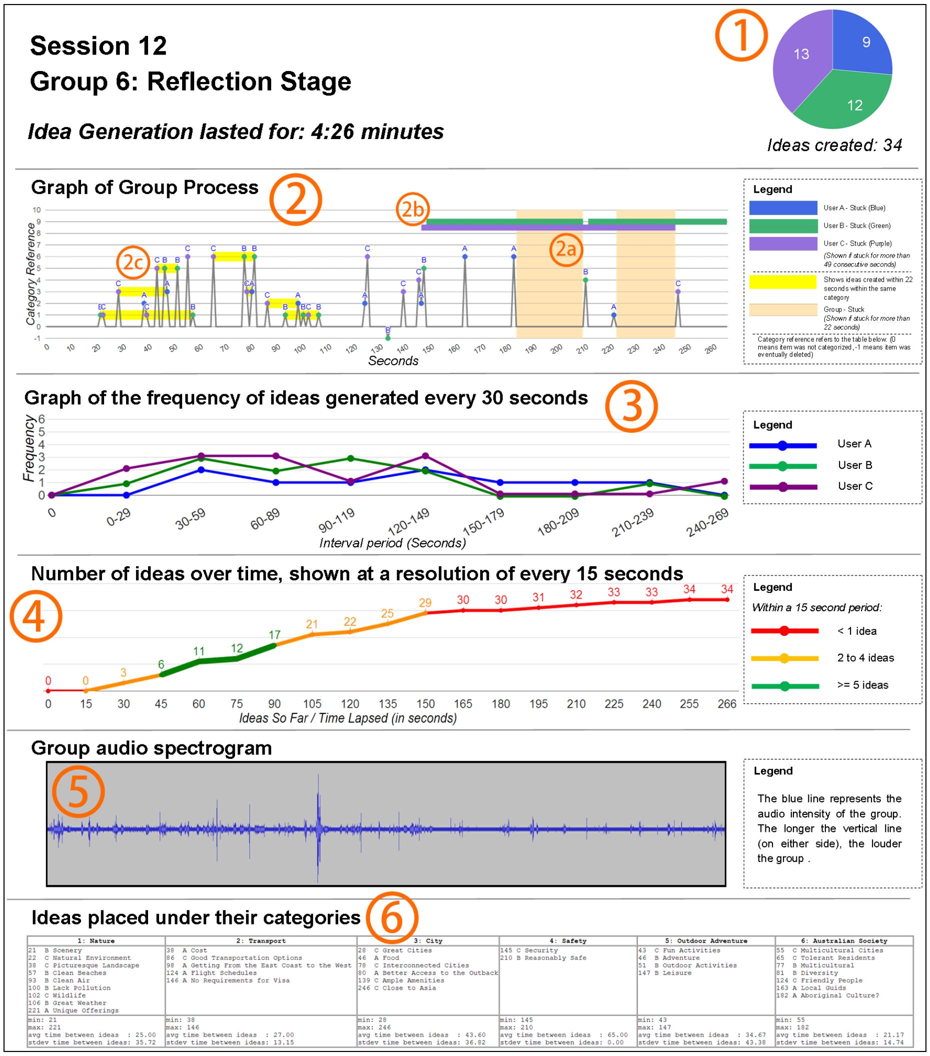


Fig. 2. Open Learner Model Visualisations.

5 Evaluation

We conducted an interview/think-aloud study with 15 participants drawn from the earlier brainstorming study (10 male, 5 female, age range 21-30, mean age 24), each interviewed separately. The study consisted of analysing 3 anonymised brainstorming sessions from the earlier study (the same 3 anonymised sessions across all interviews). The visualisations were presented on laminated A3 sheets of paper to aid visibility, and contained the different OLMs like the one shown in Figure 2 – which allowed learners to quickly point to the different items when answering the questions. These questions, listed in Table 1, investigated whether participants, could obtain information, about individual/group contributions (Q1−4), if they could identify periods when the group or its members got 'stuck' (Q5-6) or if they could define whether the group members sparked off of each other (Q7−9). Questions 10 and 11 served as self-assessment of the group and individual performance respectively. The interview questions (Table 1) linked to our core research questions as shown in Table 2. The interview process had the following steps:

Step 1 Participants were asked to pretend to be a learner that produced 13 ideas in a group who made 34 ideas (i.e. to be the purple user in Figure 2), and answer the questions in Table 1.

Step 2 Participants were shown a numerically well performing group whom created 80 ideas and asked to review their answers to Q10 and Q11. We did this to see if people would change their response, given extra information.

Step 3 Participants were asked to pretend to be a learner with 52 ideas in a group with 98 ideas, and answer the questions in Table 1.

Step 4 Participants were asked three general questions: (1) Whether they would like to see these visualisations as part of a reflection stage on a tabletop; (2) Whether they thought the visualisations would enable a group to become more effective; and (3) If you were a user with a low number of ideas, would the visualisations make you more aware and conscious about your performance.

Interview Questions		
Q1	I could work how much was my contribution	(C1)
Q2	I could figure out when we made the most ideas in the session	(C1)
Q3	I could see who created each idea	(C1)
Q4	I could see when the group was talking	(C1)
Q5	I could figure out when the group got stuck	(C2)
Q6	I could figure out when I got stuck in the session	(C2)
Q7	I could figure out the times when the group created a burst of ideas that ended out in the same category	(C3)
Q8	I could figure out periods when the group was on a roll (i.e. good sustained idea generation)	(C3)
Q9	I could see how the ideas were categorised (i.e. how ideas were grouped)	(C3)
Q10	I thought the group did a good job in the brainstorm	
Q11	I thought I did a good job in the brainstorm	

Table 1. Interview questions investigating the usefulness of the group OLMs.

	Abbreviation	Core research question	Revealed in:
(C1)	Contributions	How much did I contribute?	Q1, Q2, Q3, Q4
(C2)	Stuck	At what times was the group or an individual stuck?	Q5, Q6
(C3)	Sparking	Where did group members seem to 'spark' off each other?	Q7, Q8, Q9
	Other impact	The impact of showing learners OLMs of different groups	Q10, Q11

Table 2. Relationship between research questions and interview questions.

Responses were given on a 6-point Likert scale (1 for strongly disagree, 6 for strongly agree). Participants were instructed to point to any items (the charts/table) that influenced their response as well as provide an explanation for each item chosen. Results are summarised in Table 3.

		Contributions				Stuck		Sparking			Other impact	
Questions		Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11
Step 1 **34 idea group**	Likert	5.07	5.53	4.87	5.40	5.67	5.87	4.20	5.00	5.20	4.40	4.73
	Item	1,3	4,3	2,6	5,2	2,4	2,3	2,6	4,2	6,2	1,2	3, 2
Step 2 **80 idea group**	Likert										**3.40**	**4.40**
Step 3 **98 idea group**	Likert	5.53	4.93	5.33	5.20	5.27	5.40	**5.20**	5.20	5.27	5.20	5.60
	Item	1,6	4,2	6,2	5	2,4	2,3	2,6	4,3	6,2	1,4	1,3

Table 3. Results of the interview. Item refers to those as labelled in Figure 2, briefly: 1–pie chart; 2–graph of group process; 3–graph of frequency of ideas; 4–number of ideas over time; 5–group audio spectrogram; and 6–the table. The two most commonly referenced items are included. Bold indicates a statistically significant change from Step 1 to 2 (Q10,11) and from Step 1 to 3 (Q1-9).

Most of the learners agreed that the OLM visualisations provided key information about the group brainstorm ($\geq$4.20 across the Likert scores). While participants thought aloud, more than half mentioned ease of understandability, especially by the time they saw the third groups OLMs. Some users had initial difficulties understanding certain visualisations, for example four users initially found chart 2 to be very complex, though by the end of the activity, only two of these four still found the visualisation complex.

6 Discussion

6.1 Group members contributions to the brainstorm

In the absence of a benchmark to compare the number of ideas generated, participants determined if a group did a good job, by judging levels of equality, referring to charts 1 and 3. When additional group OLMs were introduced, participants focused on the amount of ideas produced. For individual contribution – Q1, participants drew from charts 1 and 3 and the table. Chart 1 presented overall contribution in a simple form: *P4– "easy to understand"*; *P5– "very clear"*; and *P3– "I have the biggest cut of the pie"*. Chart 3 revealed contributions over time: *P6– "I generated the most ideas in the first 90 seconds"*; and *P2– "I compared the number of ideas generated and saw that I created just as many as the others"*. For determination of active periods (Q2), 12 people (P1,2,3,4,8,9,10,12,13,15) consulted chart 4 – referencing the colour scheme. A small number of participants referred to chart 3, looking at times when frequency of ideas generated was high across all members. For whom created each idea – Q3, chart 2 and the table were referenced. For chart 2 – the coloured dots representing authors were used (P1,5,7,8,9,10,11), and for chart 6 – the author written alongside the idea (P2,3,6,12,14). Overall, the following were referred to the most: chart 1 – for individual contribution; chart 2 – for whom created each idea; and chart 4 – for periods containing a large number of ideas.

6.2 Periods where the group or individuals got stuck

For Q5 − identify when the group was stuck and Q6 − identify when individuals were stuck, the average Likert score was above 5 (Q5: 5.70 & 5.27, Q6: 5.87 & 5.40). Participants utilised charts 2, 3 and 4. For chart 2 − the shaded regions and horizontal bars were referenced (P1,7,8,9,10,11,12,15): *P9– "I looked at the interval between ideas"*; *P3– "I looked for the shades to see if they were stuck, when I couldn't see any, so I checked this one [chart 4] to see if there were any red lines"*; and *P10– "easy to see when I was stuck, because of the highlights"*. For chart 3 – participants looked for when groups tapered off, shown as dips (P1,2,3,4,6,9,14): *P2– "The graph plateaued at the end, showing me they got stuck"*, similarly in chart 4 – the gradient of the line combined with the colour coded segments (P4,5,9,11,13): *P5– "because of the red"*. Overall, chart 2 proved to be most useful for identifying stuck periods. These observations reinforce the usefulness of the information added from our brainstorming model, in providing potentially useful visual indicators to learners. These indicators (the shading, bars and coloured segments) can be the basis for discussion, reflecting on actions that led to identified periods of inactivity.

6.3 Evidence that group members 'sparked' off of each other

Question 7 asked whether a burst of ideas ended up in the same category. For this question, chart 2 was referenced, but with mixed responses. 8 participants said the yellow highlight in chart 2 was obvious: *P13– "I looked at the yellow lines, as it easily caught my attention"*, but 4 participants did not find the highlight obvious and instead horizontally scanned the grey line present on each row. Three participants mentioned the table, and said that if they spent more time they could of worked out which ideas from whom sparked other ideas, but were off put by the presentation, being heavy in text, compared to the other items. Determining when a large number of ideas was created, without the constraint of them being in the same category, participants shifted focus to chart 4. Overall, chart 2 was most useful for showing when members sparked off of each other. This can be used as a starting point for discussion in a reflection stage to talk about sparking and what led to it, and how often it occurred.

6.4 The impact of showing learners OLMs of different groups

Participants were shown an example of a particularly productive group after the first group and asked to reflect on Q10 and Q11, questions which related to performance. For group performance (Q10), upon seeing another group, with a higher number of ideas, 8 people (P2,3,7,9,10,13,14) downgraded their answer with an average reduction of 2 Likert points, resulting in a statistically significant decrease (from 4.4 to 3.4), representing a switch from the agree to the disagree side of the Likert scale. The primary reason cited was the difference in the number of ideas created (P2,3,7,9,10), and the lack of stuck periods in the new group (P13,14). Three participants (P11,12,15) kept their original answer stating whether a group performed well is more complex than a numerical figure, raising issues of group dynamics, questions about quality, and requested other group OLMs to have more information to compare against: *P12– "I only have 2 groups to go off, not a complete average, also I don't know if their quality was the same"* and *P7– "The first group generated longer multiple word ideas, while this group created single word ideas, I think that's why the first group had less ideas"*. For Q11, 5 participants changed their response, with the bulk of participants pointing out that the user with 13 ideas (the purple user) made the most ideas of the group (P1,4,8,9,11,13,15); and *P9– "purple did a good job in his group, and his performance is also dependent on his team members, so I decide to keep my original answer the same"*. Two participants (P6,11) mentioned they wanted to have an average value, to put the number of created ideas into perspective.

These comparisons point to the fact that participants are not only influenced through their own contributions within a group, but also the performance of related groups brainstorming. An apparent strong feeling of success can be changed when exposed to other group OLMs. This is helpful in promoting reflection, in order to promote a deeper understanding of performance, and also possibly to inspire learners to develop skills to improve themselves.

Overall, the impact of showing different group OLMs was helpful with participants commenting on the use of charts 1 and 3 for individual performance and charts 2, 4 and 5 for group performance. Comments: *P12– "It gives good ideas of how their process was, and this is good for feedback which is important and it also gives a summary of what we did, and the graphs are cool to look at"*; *P13– "Users might be interested to see how they performance and if they worked together, self-reflection is really useful"*; and *P14– "It can tell users a lot of information and may help them next time and [identifying] who is least active might be encouraging to try to do better"*.

7 Conclusion

We built a series of OLM visualisations for the purpose of analysing whether individuals could understand group and individual processes in order to support reflection in group brainstorming. Results showed learners found the OLMs relatively easy to comprehend and were able to answer our core questions. In the process of the study, we learnt which visualisations were most commonly referred to and why, leading to a greater understanding of the importance of different views for reflection. Our future work will be to build this into our tabletop brainstorming system, and show the visualisations through a scripted approach, to determine the effects of the OLMs when in real use.

References

1. Al-Qaraghuli, A., Zaman, H., Olivier, P., Kharrufa, A., Ahmad, A.: Analysing tabletop based computer supported collaborative learning data through visualization. Visual Informatics: Sustaining Research and Innovations pp. 329–340 (2011)

2. Bull, S., Britland, M.: Group Interaction Prompted by a Simple Assessed Open Learner Model that can be Optionally Released to Peers. In: Proceedings of Workshop on Personalisation in E-Learning Environments at Individual and Group Level, User Modeling (2007)
3. Bull, S., Vatrapu, R.: Supporting collaborative interaction with open learner models: Existing approaches and open questions. Computer Supported Collaborative Learning (2011)
4. Bull, S., Kay, J.: Student Models that Invite the Learner In: The {SMILI} Open Learner Modelling Framework. IJAIED, International Journal of Artificial Intelligence 17(2), 89–120 (2007)
5. Bull, S., Kay, J.: Metacognition and Open Learner Models. In: The 3rd Workshop on Meta-Cognition and Self-Regulated Learning in Educational Technologies, at ITS2008 (2008)
6. Clayphan, A., Collins, A., Ackad, C., Kummerfeld, B., Kay, J.: Firestorm: a brainstorming application for collaborative group work at tabletops. In: Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces. pp. 162–171. ACM (2011)
7. Clayphan, A., Kay, J., Weinberger, A.: Enhancing brainstorming through scripting at a tabletop. In: Educational Interfaces, Software, and Technology 2012: 3rd Workshop on UI Technologies and Educational Pedagogy (2012)
8. Clayphan, A., Kay, J., Weinberger, A.: ScriptStorm: scripting to enhance tabletop brainstorming. In: Special Issue on EIST in Personal and Ubiquitous Computing (PUC) Journal (to appear) (2013)
9. Diehl, M., Stroebe, W.: Productivity loss in brainstorming groups: Toward the solution of a riddle. Journal of Personality and Social Psychology; Journal of Personality and Social Psychology 53(3), 497 (1987)
10. Do-Lenh, S.: Supporting Reflection and Classroom Orchestration with Tangible Tabletops. Ph.D. thesis, École Polytechnique Fédérale De Lausanne (2012)
11. Isaksen, S.: A review of brainstorming research: Six critical issues for inquiry. Creative Research Unit, Creative Problem Solving Group-Buffalo (1998)
12. Mabbott, A., Bull, S.: Alternative views on knowledge: presentation of open learner models. In: Intelligent Tutoring Systems. pp. 131–150. Springer (2004)
13. Martínez, R., Collins, A., Kay, J., Yacef, K.: Who did what? who said that?: Collaid: an environment for capturing traces of collaborative learning at the tabletop. In: Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces. pp. 172–181. ACM (2011)
14. Martinez, R., Wallace, J., Kay, J., Yacef, K.: Modelling and identifying collaborative situations in a collocated multi-display groupware setting. In: Artificial Intelligence in Education. pp. 196–204. Springer (2011)
15. Martinez Maldonado, R., Kay, J., Yacef, K., Schwendimann, B.: An Interactive Teacher's Dashboard for Monitoring Groups in a Multi-tabletop Learning Environment. In: Intelligent Tutoring Systems. pp. 482–492. Springer (2012)
16. Osborn, A.: Applied Imagination, principles and procedures of creative thinking. Scribner's (1953)
17. Schon, D.: The reflective practitioner: How professionals think in action. Basic Books (1983)
18. Wang, H., Li, T., Rosé, C., Huang, C., Chang, C.: Vibrant: A brainstorming agent for computer supported creative problem solving. In: Intelligent Tutoring Systems. pp. 787–789. Springer (2006)

Encouraging Online Student Reading with Social Visualization Support

Julio Guerra, Denis Parra, Peter Brusilovsky

School of Information Sciences
University of Pittsburgh
130 N Bellefield Ave, Pittsburgh, PA, 15260, USA
{jdg60,dap89,peterb}@pitt.edu

Abstract. In this paper, we describe ReadingCircle, a system designed to explore an alternative approach to encouraging reading among students. It is based on recent research on open student modeling, social comparison and social visualization. The idea of this approach is to develop social visualization of students' reading progress. The visualization will reveal such reading progress through several levels (from chapters to sections to pages) and allow students to visually compare their progress with both the class as a whole and individual peers.

Keywords. elearning, online reading, social visualization, social comparison, open student model

1 Introduction

Almost every college course requires students to complete weekly readings from course textbooks or other course materials, an effort critical to the students' success in the course. However, it is not easy for an instructor to determine whether or not the students have in fact completed the assigned readings. To combat this trend, instructors have to implement various approaches to encourage student reading and to ensure that reading assignments are completed. In smaller classes, these approaches could be both creative and efficient – such as group discussions. In larger classes, however, instructors find it difficult to assess the students' progress on the readings in an efficient way. Contemporary approaches such as randomly surveying students in class or administering pop-quizzes are neither creative nor efficient. Also, reading assignments produce no artifacts to grade by. As a result, the students frequently are not motivated to complete the reading assignments.

In this paper, we describe *ReadingCircle*, an alternative approach to encouraging student reading that is based on our recent research combining open student modeling, social comparison and social visualization [1]. The premise of this approach is to engage social visualization of student reading progress as a barometer of progress. The visualization exhibits progress on several levels (from chapters to sections to pages), and allows the students to visually compare their progress with both the class as a whole and individual peers. We expect that social progress visualization will improve student awareness of readings left to do and class progress; the ultimate goal is to encourage students to do more readings. This paper presents our motivation for designing and creating this social reading application.

2 Related Work

Social Comparison. According to social comparison theory [7], people tend to compare their achievements and performance with others who are similar to them in some way. Earlier social comparison studies [11] demonstrated that students were inclined to select the more challenging tasks because of being exposed to social comparison conditions. Later studies showed that social comparison decreases social loafing and increases productivity by reinforcing good behavior through a graphical feedback tool [9]. A synthesis review of social comparison studies' summarized that applying social comparison in the classroom often leads to better student performance [8].

Social Visualization in E-Learning. The visual approach is a common technique to represent or organize data about multiple students in an informative way. For instance, social navigation, which is a set of methods for organizing users' explicit and implicit feedback to support information navigation [5], leverages the social phenomenon where people tend to follow the "footprints" of other people [2]. The educational value of social navigation have been confirmed in several studies [3, 6]

It is common to provide learners with the average values of the group model through social visualization in E-Learning; such as the average knowledge of the group on a given topic. Vassileva and Sun [10] investigated community visualization in online communities. They opined that social visualization increases social interaction among students, encourages competition, and offers students the opportunity to build trust in others and in the group. Bull & Britland [4] showed that releasing the models to their peers increases the discussion among students and encourages them to start working with learning content sooner.

In our prior work [1] we combined social visualization with open student modeling visualization to provide students with a holistic and easy-to-grasp view of their progress on answering java programming questions, and at the same time, allowing them to compare their progress with that of other students in the class. Our classroom studies demonstrated that the social visualization interface provided a remarkable increase in student work with problems. It also demonstrated that a circular design provides a better approach than a tree map to show progress over hierarchically structured content. This paper extends this work and presents a *social progress visualization interface to support online reading*. This interface takes advantage of some of the successful design ideas from our previous projects, and aims to work with a very different type of content. We expect that the new interface will provide clear guidance to the students to manage their reading process and to significantly increase their motivation to read.

3 The ReadingCircle Interface

The main challenge in our social reading interface design was to combine a simple social progress visualization of student progress over a flat list of topics (our past interface works with topics in Java) with a more complicated and hierarchical structure of student reading assignments. In addition, we wanted to employ the visualization not only as a social comparison tool, but also as a social navigation tool that provides orientation support and navigation support for a large body of assigned readings.

In light of these goals, the ReadingCircle system interface is divided into a social navigation component and a reading element as can be seen in Figure 1. The reading part on the right shows the current reading material and allows the student to make annotations and see annotations from peers. The social navigation component on the left aims to present visually the open student and peer models. The visualization of the student model (the top right part in Figure 1) is also the main content navigation control. We chose this circular shape approach because it requires less space to show the whole (hierarchical) content structure.

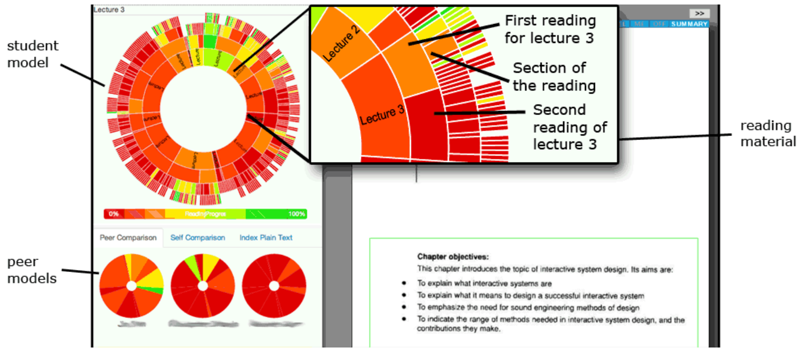


Figure 1. The ReadingCircle interface. The left part shows the student model (top) and peer models (bottom). The material is shown on the right side. A small portion of the user model is magnified at top center.

The circular shaped model presents the content structure of a course, organized clockwise, of 13 lectures. Each lecture consisted of one or more readings which can be chapters or sections from several books used in the course. Following the hierarchical structure of the reading (for example, a chapter has sections, and sections has subsections), the sector in the visualization corresponding to the reading is "opened" to reveal the fine-grained content. The top center rectangle in Figure 1 presents a closer view of the third lecture (lecture 3). By clicking in each sector, the student is presented with a menu of the related content displayed in the right side. The color of the sections indicates the progress on a scale ranging from red (not seen) to green (completed). The progress is computed by aggregating the evidence of the user reading each terminal subsection to upper level subsections, chapter and lectures. We track the individual page loads (i.e. the individual pages of each reading), and the actions (clicks, annotations) of the user in the reader interface. The bottom part of the left side in Figure 1 presents 3 tabs: Peer Comparison, Self Comparison and Index Plain Text. The Peer Comparison tab shows thumbnail models of three peers. The models display only the lecture level. The Self-Comparison tab is similar and shows three previous models of the current student (over the past 3 weeks). We aim to explore the effect of self-comparison as we study peer comparison.

The social reading interface presented above is currently going through a classroom study in a large graduate class. Using log analysis and questionnaires, we hope to assess the impact of, and the student attitude towards, the tool.

4 References

1. Hsiao, I-H., Guerra, J. , Parra, D., Bakalov, F., König-Ries, B, and Brusilovsky, P. Comparative social visualization for personalized e-learning. In Proceedings of the International Working Conference on Advanced Visual Interfaces (AVI '12). ACM, New York, NY, USA, 303-307 (2012).
2. Brusilovsky, P., Chavan, G., and Farzan, R. Social adaptive navigation support for open corpus electronic textbooks. In: Proc. of Third International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH'2004), Springer-Verlag, 24-33 (2004)
3. Brusilovsky, P., Sosnovsky, S., and Yudelson, M. Addictive links: The motiva-tional value of adaptive link annotation. New Review of Hypermedia and Multimedia, 15, 1 (2009), 97-118.
4. Bull, S. and Britland, M. Group Interaction Prompted by a Simple Assessed Open Learner Model that can be Optionally Released to Peers. In Proc. of Workshop on Personalization in E-learning Environments at Individual and Group Level at the 11th International Conference on User Modeling, UM 2007, 24-32 (2007).
5. Dieberger, A., Dourish, P., Höök, K., Resnick, P., and Wexelblat, A. Social navigation: Techniques for building more usable systems. interactions, 7, 6 (2000), 36-45.
6. Farzan, R. and Brusilovsky, P. AnnotatEd: A social navigation and annotation service for web-based educational resources. New Review in Hypermedia and Multimedia, 14, 1 (2008), 3-32.
7. Festinger, L. A theory of social comparison processes. Human Relations, 7 (1954), 117-140.
8. Pieternel Dijkstra, Hans Kuyper, Greetje van der Werf, and, A.P.B., and Zee, Y.G.v.d. Social Comparison in the Classroom: A Review. REVIEW OF EDUCATIONAL RESEARCH, 78, 4 (2008).
9. Shepherd, M.M., Briggs, R.O., Reinig, B.A., Yen, J., and Jay F. Nunamaker, J. Invoking social comparison to improve electronic brainstorming: beyond anonymity. J. Manage. Inf. Syst., 12, 3 (1995), 155-170.
10. Vassileva, J. and Sun, L. Evolving a Social Visualization Design Aimed at Increasing Participation in a Class-Based Online Community. International Journal of Cooperative Information Systems (IJCIS), 17, 4 (2008), 443-466.
11. Veroff, J. Social comparison and the development of achievement motivation., New York: Sage (1969).

Academically Productive Talk: One Size Does Not Fit All

David Adamson & Carolyn P. Rosé

Language Technologies Institute, Carnegie Mellon University, Pittsburgh PA, 15213
{dadamson,cprose}@cs.cmu.edu

Abstract. We present a study in which we experimentally manipulate the form of support offered to groups of three students during collaborative learning. Specifically, we contrast two forms of Academically Productive Talk (APT) facilitation, known as Revoicing and Agree-Disagree. The first form has been demonstrated effective with the target age group (i.e., 9th grade) on an earlier more difficult unit. The second form has been demonstrated effective with older kids. Results suggest that with this age group, facilitation with Revoicing may be more effective than Agree-Disagree. Implications for future work are discussed.

Keywords: dynamic support for collaborative learning, academically productive talk, discussion for learning.

1 Introduction

Collaborative learning activities, when delivered effectively, can provide significant cognitive, metacognitive, and social benefits to students [18][32][35]. Studies in the field of computer-supported collaborative learning have demonstrated the pedagogical value of social interaction [37][38]. Prior work on adaptive support for collaborative learning has adapted hint-based support originally developed for individual learning to support peer tutoring [13], and other work has grown out of earlier efforts to develop tutorial dialogue agents originally designed for individual learning [16][30][40][41]. This form of dynamic agent-based support for collaborative learning was historically tailored to specific learning populations and content domains [22], which limits its generality. More generalizable forms of support would increase the potential for impact, but as we discuss in this paper, raise new questions about principles for adaptation that would enable us as system developers to provide solutions that can be effective for diverse student populations.

Our recent efforts are in the direction of intelligent conversational agents acting as discussion facilitators, offering support behaviors that are not tied to a particular content-area or context [1][10][14]. The design of such support is in line with the literature on facilitation of collaborative learning groups [17]. In particular, it draws upon a body of work that has shown that certain forms of classroom discussion facilitation, termed Accountable Talk, or Academically Productive Talk (APT), are beneficial for learning with understanding [3][8][9][28][29][33][34][39].

In this paper we present results from a study in which we contrast two forms of APT based support. The first form, Revoicing support, has been found in prior work to achieve positive learning effects with the target student population of 9th graders [14] on an earlier and more difficult lesson. The other form of support, Agree-Disagree support, has been found to be effective with older, more advanced learners [1] in a different content domain. In this study, we show that with a 9th grade student population, Revoicing support is slightly more effective that Agree-Disagree support. These results contribute towards an empirical foundation for adapting APT based support to differences in content domain difficulty and differences in the developmental stage of target learners.

In the remainder of the paper we first review the state of the art in agent based support for collaborative learning. Next we describe two forms of APT-based support. Then we describe an evaluation study where we compare the effectiveness of these two forms of support for 9th grade biology students working on a genetics unit that is relatively easy for them. We conclude with discussion of results and future directions.

2 Prior Work

Academically Productive Talk has grown out of frameworks that emphasize the importance of social interaction in the development of mental processes. Michaels, O'Connor and Resnick [26] describe a number facilitating moves that teachers can employ to promote student-centered classroom discussion. A selection of these moves are presented in Table 1. In studies where teachers used similar facilitation strategies, students showed dramatic improvement on standardized math scores, transfer to reading test scores, and retention of transfer for up to 3 years [8][9].

Table 1. Selected Accountable Talk Moves

APT Move	Example
Revoicing a student's statement	"So, let me see if I've got your thinking right. You're saying XXX?"
Asking students to apply their own reasoning to someone else's reasoning	"Do you *agree or disagree*, and why?"

Collaboration scripts are a common way to describe and structure support for collaborative learning [20] within the field of computer-supported collaborative learning. A collaboration script may describe any of a wide range of features of collaboration scenarios, including the tasks, timing, roles, and the methods and desired patterns of interaction between the participants. A script can describe the collaborative activity at the macro or micro level [12]. Macro-scripts describe the sequence and structure each phase of a group's activities, specifying coarse-grained features such as assigned tasks and roles, and the overall shape of the activity. Micro-scripts, on the other hand, are models of dialogue and argumentation embedded in the activity, and are intended to be adopted and progressively internalized by the participants [19]. Micro-scripts can be realized by sharing prompts or hints with the user, guiding or providing models for their contributions [36]. While traditional collaboration scripts such as these can provide some degree of support for conversational and reasoning practices, they fall short of delivering the active, engaged facilitation described by the APT literature.

In particular, such scripts are static, and do not respond to changes in (or awareness of) student need or ability during the activity. Such non-adaptive approaches risk detrimental over-scripting [11]. More preferable would be the delivery or adjustment of supports in response to the automatic analysis of student activity [2][31]. The collaborative conversational agents described by Kumar and Rosé [24] were among the first to implement such dynamic scripting in a CSCL setting, with demonstrable gains over otherwise equivalent static support. Likewise, recent work by Baghaei et al [6] and Diziol et al [13] show that adaptive supports can have meaningful effects on student learning and interaction.

3 Dynamic Support for Academically Productive Talk

Two dynamic conversational supports based upon APT facilitation, namely Revoicing and Agree-Disagree, were implemented and evaluated in this study. The open-source Bazaar architecture [2] was used to author and orchestrate the conversational agent and the support behaviors described below.

3.1 Revoicing Support

One of the forms of support evaluated in this paper is a Bazaar component that performs an Academically Productive Talk move referred to as Revoicing. The agent compares student statements against a list of conceptually correct statements developed with teachers. In the study described in this article, 35 such statements were developed and validated against pilot data. For each student turn, we calculate a measure of “bag of synonyms” cosine similarity against each expert statement, based on the method described by Fernando and Stevenson [15]. If this similarity value exceeds a conservatively high threshold, we consider the student's turn to be a possible paraphrase of the matched statement, and thus “revoicable” (this threshold was determined through tests against pilot data, such that at least 80% of the revoicings suggested for candidate student were on-target). The Revoicing component may respond by offering the matched statement as a paraphrase of the student's turn, for example “So what I hear you saying is XXX. Is that right?” No statement may trigger a revoice move more than once.

3.2 Agree-Disagree Support

The other support we evaluate is a Bazaar component which performs the APT Agree-Disagree move. Candidate student statements are identified using the same method as described for the Revoicing support, but with a lower threshold that allows looser matches. After detecting such a candidate, the agent waits for the other students in the group to respond to it. If another student responds with an evaluation of their peer’s contribution (for example, “I agree” or “I think you’re wrong”, as recognized by a small list of hand-crafted regular expressions), but doesn’t support the evaluation with an explanation, the agent will encourage this second student to provide one. If a student instead follows up with another APT candidate statement, the agent does nothing, leaving the floor open for productive student discussion to continue unimpeded, reducing the risk of over-scripting their collaboration. If the other students do not respond with either an evaluation or a contentful follow-up, the agent prompts them to comment on the candidate statement – for example, “What do you think about Billy’s idea? Do you agree or disagree?”

4 Method

Following the literature on APT used as a classroom facilitation technique, in this study we test the hypothesis that appropriate APT support in a computer-supported collaborative learning setting will both intensify the exchange of reasoning between students during the collaborative activity, and increase learning during the activity.

4.1 Instructional Content and Study Procedure

Participants: This study was conducted in seven 9^{th} grade biology classes of an urban school district. The classes were distributed across two teachers (with respectively 3 and 4 classes) for a total of 143 students total, with 76 consenting. Students were randomly assigned to groups of 3. Groups were randomly assigned to conditions. Only data from consenting students was used in the analysis presented here.

Experimental Manipulation: This study was run as a 3 condition between subjects design in which the APT agents provided some behaviors in common across conditions, but other behaviors were manipulated experimentally. Across all conditions, the agent provided the same macro level support by guiding the students through the activity using the same phases introduced in such a way as to control for time on task. It was the micro-scripting behaviors that were manipulated experimentally in order to create the three conditions of the design. The first experimental conditions was Revoicing, using the behavior described above. The second was the Agree-Disagree condition, where the Agree-Disagree behavior discussed above was used. In the control condition, neither of these behaviors was used.

Learning Content: The study was carried out during a module introducing the concepts of genetics, heredity, and single-trait inheritance. In the activity, student groups were presented with a set of three problems and asked to reason about the physical and genetic traits of the likely parents of a set of siblings. Specifically, in each problem, students were shown a litter of eight kittens that varied in fur color (either orange or white), and were instructed to identify the genotypes and phenotypes of the parents, and to explain their reasoning to their teammates. This sort of “backwards” reasoning had not been explicitly addressed in the course to date – students only had prior experience with “forward” reasoning from given parental traits. The mystery parents were presented as the inputs to an unpopulated Punnett square, as shown in Figure 2. As an incentive, students were told that the best team, determined by a combination of discussion quality and post-test scores, would be awarded a modest prize. Each of the three tasks was progressively harder than the last in that fewer clues about the parent's identities were included. The collaborative task content, the macro-scripts that supported it, and the list of statements powering the APT support were all developed iteratively with feedback from teachers and content experts.

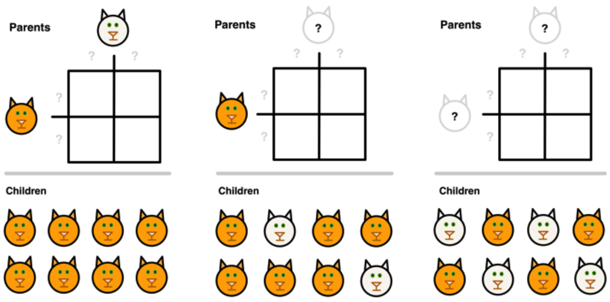


Fig. 1. Task sequence for the collaborative activity.

Study Procedure: The study was conducted over three phases, which occurred as single class periods over two school days. The first phase ("day 1") involved the teachers taking a pre-test at the end of a regular class session.

The second phase ("day 2") was centered around a 20 minute collaborative computer-mediated activity during which the experimental manipulation took place. The students performed the activity in groups of three, scaffolded by a conversational agent. Students within classes were randomly assigned to groups, then groups to conditions. The activity was introduced by a cartoon handout depicting the use of APT, and a ten-minute presentation describing the task and reviewing the basics of genetics and heredity. At the end of this second phase, the students took a post-activity test.

The computer activity was intended to equip the students with enough empirical data and attempts at reasoning to prepare them for the third phase ("day 3"), a full class APT discussion with their teacher, during which they would reconcile their different understandings and explanations. At the end of this discussion, they took a post-discussion test.

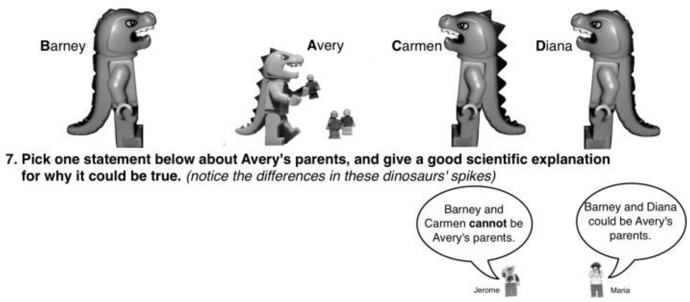


Fig. 2. Concept cartoon question from the post-activity test.

4.2 Measurement

Domain knowledge was measured at three time points using a paper based test. Each of the three tests (Pre-Test, Post-Activity Test, Post-Discussion Test) followed a similar format: a set of multiple choice problem-solving questions addressing forward and backward reasoning about single inheritance, and what we refer to as a *concept cartoon*, in which a set of potential parents for a single child was displayed, along with two hypotheses for who the child's parents might be. Students were instructed to select one hypothesis and clearly explain the conditions that would allow it to be true – either hypothesis could be correct, with different underlying assumptions. Student responses were graded with a rubric assessing the quality and depth of their explanation, including explicit displays of reasoning.

Each test covered the same knowledge but used different scenarios. The knowledge to be covered by each test was established in coordination with the teachers, with teacher trainers who identified common misconceptions, and with test results from a study run with the same content the previous year. After an initial round of consensus grading by two graders on a subset of the tests to establish a scoring guide, the remaining tests were divided and scored by one grader each.

Table 2. Total test scores (standard dev) for Pretest, Post-Activity Test, and Post-Discussion Test in the 3 conditions.

	Control	Revoice	Agree-Disagree
Pretest	5.5 (3.1)	5.5 (3.2)	3.9 (3.0)
Post-Activity Test	6 (3.4)	6.3 (3.1)	4 (3.1)
Post-Discussion Test	5.7 (3.1)	6.1 (2.9)	4.8 (3.3)

4.3 Results

First we tested whether students learned during the online activity. Test scores were divided into explanation questions and problem solving questions. Thus, for each test, each student has two scores. In order to evaluate learning, we used an ANOVA with Test Score as the dependent variable, Explanation vs Skill, Pretest vs Post-Activity Test, Condition, and Teacher as independent variables. We added Teacher as a variable because we noticed that students from one teacher learned significantly more than students from the other teacher. In this analysis, all of the independent variables were significant except Pre-test vs Post-test, which was marginal, $F(1, 270) = 3.6$, $p < .06$. There were no significant interactions between independent variables. Thus we find qualified evidence that students learned during the online activity, across conditions. However, on inspecting the average scores in Table 1, we see barely any evidence of learning in the Agree-Disagree condition. The most learning we see is about .25 standard deviations in the Revoicing condition, and about half that in the Control condition.

We also tested whether students learned during the Post-activity discussion. In this case, when comparing between the Post-Activity test and the Post-Discussion test there was no significant difference. In fact, the trend was that students scored more poorly on the Post-Discussion test than the Post-Activity test, except in the Agree-Disagree condition, where the students came into the discussion with less knowledge than students in the other two conditions, and seemed to be able to use the Post-activity Discussion to catch up, which is consistent with findings from earlier studies (Dyke et al., in press).

We compared learning across conditions between Pre-test and Post-Activity test, and between Pre-test and Post-Discussion test. In both cases, we used an ANCOVA with the posttest measure (i.e., Post-Activity test in the first comparison and Post-Discussion test in the second) as the dependent variable and the Pre-test as the covariate. We retained the Teacher variable in addition to the condition variable. In neither case do we find a significant effect of condition. However between the Pre-test and Post-activity test the trend is for adjusted posttest scores to be higher than the control condition in the Revoicing condition (by .13 standard deviations) and lower than the control condition in the Agree-Disagree condition (by .4 standard deviations), with very similar trends when comparing between Pre-test and Post-Discussion test.

We acknowledge that stronger claims could be made by conducting our analysis using multilevel modeling. However, such complex modeling techniques require larger data sets in order to avoid falling prey to type II errors during hypothesis testing. Due to the small size of our data, we employed simpler methods for our analysis.

5 Discussion & Conclusions

Overall, the results are weak. However, the results suggest a differential effect of the two experimental conditions. The trend in favor of the Revoicing condition is consistent with earlier studies with the same age group, but on a more difficult unit in the course [14]. The trend to learn less than the control condition in the Agree-Disagree condition is in contrast to earlier results with more advanced learners [1] where students in the Agree-Disagree condition learned significantly more than in the control condition. These suggestive results will need to be followed up with additional experimentation before we can have more confidence in the findings. However, they do suggest that the effect of these APT facilitation strategies on learning depend on the difficulty of the unit and the developmental stage of the learners, and that more results are needed to inform effective strategies for supporting groups of learners.

6 Acknowledgements

This work was supported in part by NSF grant SBE 0836012 to the Pittsburgh Science of Learning Center.

7 References

[1] Adamson, D., Ashe, C., Jang, H., Yaron, D., & Rosé, C. (2013). Intensification of Group Knowledge Exchange with Academically Productive Talk Agents. *Proceedings of the 10th International Conference on Computer Supported Collaborative Learning*, Madison Wisconsin, July 2013.

[2] Adamson, D., & Rosé, C. (2012). Coordinating multi-dimensional support in collaborative conversational agents. In *Proceedings of Intelligent Tutoring Systems* (pp. 346-351). Springer Berlin/Heidelberg.

[3] Adey, P., & Shayer, M. (1993). An Exploration of Long-Term Far-Transfer Effects Following an Extended Intervention Program in the High School Science Curriculum. *Cognition and Instruction*, *11*(1), pp 1-29.

[4] Ai, H., Kumar, R., Nguyen, D., Nagasunder, A., Rosé, C. P. (2010). Exploring the Effectiveness of Social Capabilities and Goal Alignment in Computer Supported Collaborative Learning, in *Proceedings of Intelligent Tutoring Systems.*

[5] Azmitia, M. & Montgomery, R. (1993). Friendship, transactive dialogues, and the development of scientific reasoning, *Social Development* 2(3), pp 202-221.

[6] Baghaei, N., Mitrovic, A., & Irwin, W. (2007). Supporting collaborative learning and problem solving in a constraint based CSCL environment for UML class diagrams, *International Journal of Computer Supported collaborative Learning* 2(3), pp 159-190.

[7] Berkowitz, M., & Gibbs, J. (1983). Measuring the developmental features of moral discussion. *Merrill-Palmer Quarterly*, 29, pp 399-410.

[8] Bill, V. L., Leer, M. N., Reams, L. E., & Resnick, L. B. (1992). From cupcakes to equations: The structure of discourse in a primary mathematics classroom. *Verbum*, 1, 2, 63-85.

[9] Chapin, S., & O'Connor, C. (2004). Project challenge: Identifying and developing talent in mathematics within low-income urban schools. *Boston University School of Education Research Report* (Vol. 1, pp. 1-6).

[10] Clarke, S., Chen, G., Stainton, K., Katz, S., Greeno, J., Resnick, L., Howley, H., Adamson, D., Rosé, C. P. (2013). *The Impact of CSCL Beyond the Online Environment, Proceedings of Computer Supported Collaborative Learning*

[11] Dillenbourg, P. (2002). Over-scripting CSCL : The risks of blending collaborative learning with instructional design . *Three worlds of CSCL: Can we support CSCL*, pp. 61-91.

[12] Dillenbourg, P., Hong, F. (2008). The mechanics of CSCL macro scripts. *International Journal of Computer-Supported Collaborative Learning* 3(1), pp 5-23.

[13] Diziol, D., Walker, E., Rummel, N., & Koedinger, K. R. (2010). Using intelligent tutor technology to implement adaptive support for student collaboration. Educational Psychology Review, 22(1), 89-102.

[14] Dyke, G., Adamson, D., Howley, I., Rosé, C.P. (Under Review). Enhancing Scientific Reasoning and Explanation Skills with Conversational Agents, submitted to *IEEE Transactions on Learning Technologies*.

[15] Fernando, S., and Stevenson, M. (2008). A semantic similarity approach to paraphrase detection, *Computational Linguistics UK (CLUK 2008) 11th Annual Research Colloquium.*

[16] Graesser, A., VanLehn, K., the TRG, & the NLT (2002). Why2 Report: Evaluation of Why/Atlas, Why/AutoTutor, and Accomplished Human Tutors on Learning Gains for Qualitative Physics Problems and Explanations. *LRDC Tech Report, University of Pittsburgh.*

[17] Hmelo-Silver, C. E. & Barrows, H. S. (2006). Goals and Strategies of a Problem-based Learning Facilitator. *The Interdisciplinary Journal of Problem Based Learning*, 1(1), pp 21-39.

[18] Kirschner, F., Paas, F., & Kirschner, P. A. (2009). A cognitive load approach to collaborative learning: United brains for complex tasks. *Educational Psychology Review*, 21, 31–42.

[19] Kobbe, L., Weinberger, A., Dillenbourg, P., Harrer, A., Hämäläinen, R., Häkkinen, P., Fischer, F. (2007). Specifying computer-supported collaboration scripts. *The International Journal of Computer-Supported Collaborative Learning* 2(2-3), pp 211-224.

[20] Kollar, I., Fischer, F., Hesse, F.W. (2006). Collaborative scripts - a conceptual analysis. *Educational Psychology Review* 18(2), pp 159-185.

[21] Kumar, R., Gweon, G., Joshi, M., Cui, Y., Rosé, C. P. (2007a). Supporting Students Working Together on Math with Social Dialogue. *Proceedings of the SLaTE Workshop on Speech and Language Technology in Education*

[22] Kumar, R., Rosé, C. P., Wang, Y. C., Joshi, M., Robinson, A. (2007b). Tutorial Dialogue as Adaptive Collaborative Learning Support, *Proceedings of Artificial Intelligence in Education.*

[23] Kumar, R., Ai, H., Beuth, J., Rosé, C. P. (2010). Socially-capable Conversational Tutors can be Effective in Collaborative Learning Situations, in *Proceedings of Intelligent Tutoring Systems*.

[24] Kumar, R., Rosé, C.P. (2011). Architecture for Building Conversational Agents that Support Collaborative Learning. *IEEE Transactions on Learning Technologies* 4(1).

[25] Lison, P. (2011). Multi-Policy Dialogue Management. In *Proceedings of the SIGDIAL 2011 Conference*, Association for Computational Linguistics, pp. 294-300

[26] Michaels, S., O'Connor, C., & Resnick, L.B. (2007). Deliberative discourse idealized and realized: Accountable talk in the classroom and in civic life. *Studies in Philosophy and Education*.

[27] Rabe-Hesketh, S., Skrondal, A., & Pickles, A. (2004). *GLLAMM Manual*. University of California, Berkely. U. C. Berkeley Division of Biostatistics Working Paper Series, Paper 160.

[28] Resnick, L. B., Asterhan, C. A., & Clarke, S. N. (in press). Socializing Intelligence through Academic Talk and Dialogue. *Washington, DC: American Educational Reserach Association*.

[29] Resnick, L. B., Salmon, M., Zeitz, C. M., Wathen, S. H., & Holowchak, M. (1993). Reasoning in conversation. *Cognition and Instruction*, 11(3-4), 347-364.

[30] Rosé, C. P., Jordan, P., Ringenberg, M., Siler, S., VanLehn, K., Weinstein, A. (2001). Interactive Conceptual Tutoring in Atlas-Andes, *Proceedings of AI in Education*.
[31] Rosé, C. P., Wang, Y.C., Cui, Y., Arguello, J., Stegmann, K., Weinberger, A., Fischer, F. (2008). Analyzing collaborative learning processes automatically: Exploiting the advances of computational linguistics in computer-supported collaborative learning. *The International Journal of Computer-Supported Collaborative Learning* 3(3), 237-271.
[32] Scardamalia, M., & Bereiter, C. (1993). Technologies for knowledge-building discourse. *Communications of the ACM*, 36(5), 37–41.
[33] Topping, K. J., & Trickey, S. (2007a). Collaborative philosophical enquiry for school children: Cognitive effects at 10-12 years. *British Journal of Educational Psychology*, 77(2), 271-288.
[34] Topping, K. J., & Trickey, S. (2007b). Collaborative philosophical inquiry for schoolchildren: Cognitive gains at 2-year follow-up. *British Journal of Educational Psychology*, 77(4), 787-796.
[35] Webb, N. M., & Palinscar, A. S. (1996). Group processes in the classroom. In D. C. Berliner & R. C. Calfee (Eds.), *Handbook of educational psychology* (pp. 841–873). New York: Prentice Hall.
[36] Wecker, C., Fischer, F. (2007). Fading scripts in computer-supported collaborative learning: The role of distributed monitoring. *Proceedings of the 8th iternational conference on Computer Supported Collaborative Learning*, pp. 764-772.
[37] Weinberger, A., Stegmann, K., & Fischer, F. (2007). Scripting argumentative knowledge construction: Effects on individual and collaborative learning. In C. Chinn, G. Erkens, & S. Puntambekar (Eds.), *Mice, minds, and society: CSCL 2007* (pp. 37-39). New Brunswick, NJ: International Society of the Learning Sciences.
[38] Weinberger, A., Fischer, F. (2006). A framework to analyze argumentative knowledge construction in computer-supported collaborative learning. *Computers & Education* 46(1), 71–95.
[39] Wegerif, R., Mercer, N., & Dawes, L. (1999). From social interaction to individual reasoning: an empirical investigation of a possible socio-cultural model of cognitive development. *Learning and Instruction*, 9(6), 493-516.
[40] Wiemer-Hastings, P., Graesser, A., Harter, D. and the Tutoring Research Group, (1998). The Foundations and Architecture of AutoTutor. In B. Goettl, H. Halff, C. Redfield & V. Shute (Eds.) *Intelligent Tutoring Systems: 4th International Conference* (ITS '98) (pp334-343). Berlin: Springer-Verlag.
[41] Zinn, C., Moore, J. D., & Core, M. G. (2002). A 3-Tier Planning Architecture for Managing Tutorial Dialogue. In S. A. Cerri, G. Gouardères & F. Paraguaçu (Eds.) *Proceedings of the Sixth International Conference on Intelligent Tutoring Systems, ITS 2002* (pp. 574-584). Berlin: Springer Verlag.

Towards Supporting 'Learning To Learn Together' in the Metafora platform

Manolis Mavrikis[1], Toby Dragon[2], Nikoleta Yiannoutsou[3], Bruce M. McLaren[2,4]

[1]London Knowledge Lab, Institute of Education, University of London, WC1H 0AL, UK,
m.mavrikis@lkl.ac.uk
[2]CeLTech, Saarland University, Saarbruecken 66123, Germany,
toby.dragon@celtech.de
[3]Educational Technology Lab, National and Kapodistrian University of Athens, Greece 106 79,
nyiannoutsou@ppp.uoa.gr
[4]Human-Computer Interaction Institute, Carnegie Mellon University, Pittsburgh, PA, USA
bmclaren@cs.cmu.edu

Abstract. Computer-Supported Collaborative Learning (CSCL) has been demonstrated to improve student interaction in complex collaborative learning scenarios. When orchestrated appropriately, it also provides opportunities for learning high-level social learning skills, or "learning to learn together" (L2L2), but these opportunities are often only dealt with implicitly. This paper presents work towards an intelligent system that can scaffold L2L2 across many domains by (a) offering carefully-designed message templates that encourage peers to communicate with their groups about their learning process, (b) analyzing student work and recommending a specific set of these message templates that are pertinent to their moment-by-moment interaction. We present methods by which the system can use automated analysis techniques to recognize opportunities where students might benefit from these messages, and either send the message directly or prioritize message templates for students' use.

Keywords. Computer-Supported Collaborative Learning, Exploratory Learning Environments, Learning to Learn Together, Intelligent Support of Social Interaction

1 Introduction

The Computer-Supported Collaborative Learning (CSCL) field has demonstrated that success in complex collaborative environments depends on several factors including the type of task, the learning scenario, and the collaborative skills of the students involved. When orchestrated appropriately, these types of learning scenarios provide opportunities both for domain related learning and social meta-learning, or 'learning to learn together' (L2L2). However, for this to be possible, students and teachers alike need tools to elevate their conversation beyond solely subject matter, to recognize and practice high-level collaborative learning skills (L2L2 skills) in tandem with domain skills. Beyond simply providing appropriate interaction spaces, one of the goals behind CSCL and AI in Education systems is to provide more structured guidance. This type of automated support, however, is challenging in these complex scenarios, due to the variance in domains, learning scenarios, and intricacies of interrupting collaborative learning processes at appropriate points in time. All of these considerations limit the applicability of standard techniques for providing direct feedback.

With these challenges in mind, we argue for a broader perspective on the role of both feedback and AI in such scenarios. As discussed in [1], feedback in collaborative settings can manifest in many different ways, rather than limiting intervention mechanisms to messages flowing directly from an AI analysis system to individuals. We suggest a design where students, teachers (or facilitators in general), and automated agents can all offer feedback to individuals and groups, with the support of the system. Thus, the system takes on an additional role of providing tools and scaffolding to help students offer feedback to each other (a more indirect presentation of feedback). This scaffolding can be provided through *message templates*, generic phrases that focus attention on L2L2 concepts and can be tailored to fit the specific scenario at the time of use. These message templates are available in an intuitive and easy-to-use tool that enables students to send messages to one another, or for teachers to send messages to students. Utilizing this functionality, the AI system can go beyond the traditional role (i.e., direct presentation of feedback messages), to also scaffold the users in sending messages to each other (indirect presentation). To accomplish this, the AI system can recommend the most relevant message templates at any given point in time. Key questions to address when taking this approach include:

- What kinds of messages are most likely to promote L2L2 within task-focused group work?
- How can an intelligent system be developed to understand and identify when these messages might be most effective?
- How should the system deliver these messages or encourage the users to deliver them?

The rest of the paper is organized as follows. We first describe the context where this work is situated, in particular the Metafora platform and pedagogy that is being developed in the EU-funded Metafora project [2]. In Section 2, we briefly present the system and the key components of L2L2 that it is designed to help students develop and practice. In Section 3, we describe our process for developing appropriate, generally-applicable messages, and how these so-called ‘message templates’ are made generic and available for use within the system through techniques that allow the system to recognize and automatically respond to L2L2 behaviors. To conclude, we discuss our initial findings and future plans with respect to evaluating the approach.

2 Background and relevant work

2.1 The Metafora system and project

To support the L2L2 process, the Metafora project designed a platform that includes a planning tool designed for explicating and reflecting on the group learning process. Additionally, the platform contains the LASAD discussion environment [3] for developing arguments or discussions around the topics that emerge during the collaborative process. Of course, teaching these higher-level learning skills cannot be done without grounding the work with genuinely challenging tasks that require critical thinking skills (c.f. [4],[5]). The Metafora system offers a broad range of such learning activities across math and science by providing a suite of exploratory learning environments (microworlds and simulations). All of these tools are brought together in the Metafora platform, which serves both as a toolbox and as a communication architecture to support cross-tool interoperability. As a toolbox the system provides a graphical container in which the diverse learning tools can be launched and used (the Figures in Table A.1 give an impression of the Metafora system with the platform parts on the top and left borders and the graphically integrated tools in the main panel from center to right).

2.2 L2L2 in Metafora

The Metafora platform and tools have been designed and implemented to provide support for key components of L2L2, defined through both literature review and design-based research. In the interest of space we refer the reader to ([2]; [6]) and the project deliverables (see http://www.metafora-project.org) but in brief the four L2L2 aspects are as follows:

- *Distributed leadership:* each of the group members assumes leadership, encouraging both individuals and the group to make progress towards goals on both intellectual and managerial levels.
- *Mutual engagement:* group members co-construct, discuss/argue, or seek/offer help about mutually shared artifacts.
- *Peer Assessment and Feedback:* group members constructively evaluate *the results* of work done by themselves, their peers, and their group as a whole.
- *Group Reflection:* group members consider *the process* by which they will accomplish, are accomplishing, or have accomplished their tasks.

We see in our current research efforts [2] and ongoing experimentation that this system offers an environment in which L2L2 skills can be practiced in many scenarios. However, we recognize that presenting the learning environment without further support may not promote L2L2 explicitly, especially for novice learners, as other literature also suggests (e.g. [7],[8]). The challenge of promoting L2L2 explicitly necessitates identification of the key elements of social interaction. In this way, support and reflection can target these key elements to make collaborative learning effective.

3 Promoting L2L2 through sending and recommending messages

As described earlier, our approach to L2L2 intervention and support is to provide a tool that guides and enables students to effectively interact with one another. Other research has demonstrated the potential benefits of supporting peer tutoring, (e.g. [9],[1]). Others are also taking the approach of using an AI system to recommend feedback that should be given by a human mentor [10]. We attempt to apply both of these principles to work within our L2L2 framework, where we encourage students to engage with peers by spontaneously taking on the role of mentor, providing timely feedback and initiate discussions about their learning process To enable and encourage students to engage in these activities, we developed a messaging tool that promotes students in using specific messages to engage in L2L2 and regulate their own collaboration. This tool provides students with the means to be their own facilitators, interacting with their peers or entire group as necessary. In addition, this same system provides a method for teachers and automated agents to offer similar interventions. In order to scaffold L2L2, the system offers specific speech acts, implemented as message templates, to focus students on the high-level concepts of L2L2. Creating well-targeted, supportive, and helpful message templates is crucial to the success of such an approach, and therefore we took an iterative, data-driven approach to understanding what specific speech acts might promote positive L2L2 behaviors. These speech acts, which were collected from actual student and teacher dialog, were then abstracted as message templates, applicable across the wide range of Metafora scenarios.

3.1 Sending and receiving messages

The Messaging Tool was developed to satisfy requirements that both our previous research with similar tools [11] and early pilots allowed us to identify. While providing some scaffolding for the previously mentioned reasons, the tool also had to be simple and speed up (rather than delay) interaction between students. In addition, we wanted to provide not only opportunities for reflection but also flexibility to students and the ability to adapt the messages to their specific situation and task. As such, the tool is equipped with what we refer to as *message templates,* sentences that correspond to the four L2L2 aspects and refer in a general manner both to the stages of the students' current activity, and the different tools they may be using (particularly the planning and the discussion tool).

Any group member can select one of these message templates and then potentially edit the template to adapt to the particular situation. The messages that are sent with the tool are kept for further reflection (Fig. 1, the "sent" tab). A snapshot of the tool appears below. Fig. 1 shows the tool from which messages are sent, and Fig. 2 demonstrates how the message appears for students receiving the message.

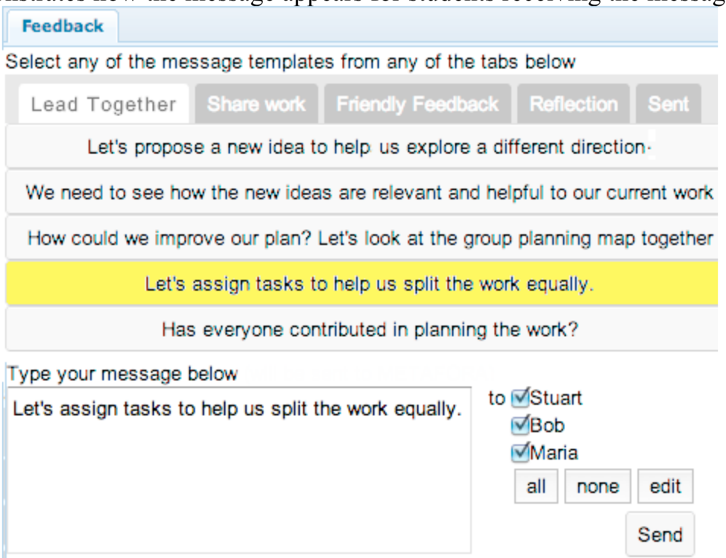


Fig. 1. The **Messaging Tool**. Students can choose and edit messages templates from each tab representing the different L2L2 aspects (the titles are adapted to children-friendly version)

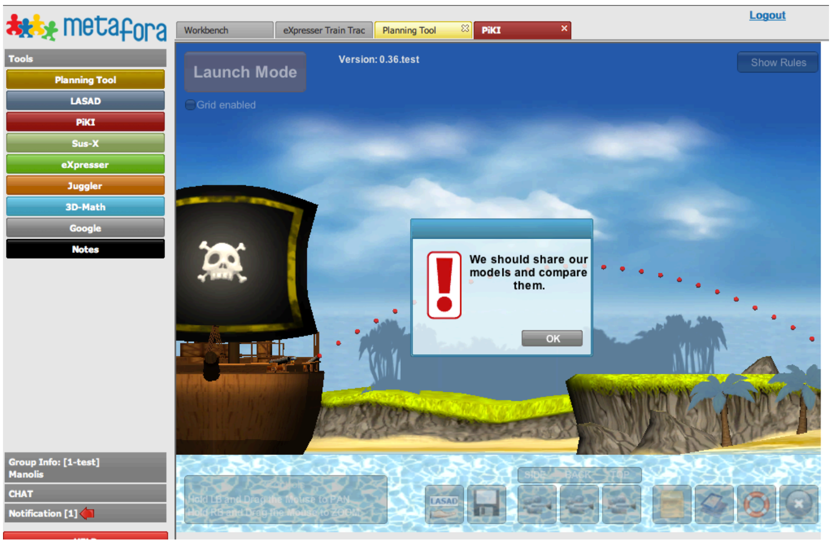


Fig. 2. Once a message is sent, it appears as pop-up anywhere that the students are working. In this case, a student is investigating their PIKI construction without much attention to the work of the rest of the group, and another student requests that they share and compare their work.

The system includes two types of message templates — peer and external — both created based on previous studies and Wizard-of-Oz experiments (c.f. [12]). Peer message templates are designed to address the group of students working together, and are sent by individual students to the rest of this group. These messages are designed to scaffold group work. External messages are equivalent messages that the system can send (whenever appropriate) as interventions. This list of 'external' messages can also be used by a teacher or any facilitators, who can launch the system separately and use it to support the students, as described in [12] where we presented similar work using these tools to simulate the provision of messages). Table A.2 in the appendix presents a tentative sample list of message templates.

3.2 Delivering and Recommending Messages

In early experimentation we observed a potential limitation of the messaging tool, in that it was challenging to identify quickly the most relevant L2L2 aspect and message templates. Taking into account that reflection is better encouraged when in context, we designed the system for highlighting (recommending) pertinent messages based on students' recent work.

This recommendation relies on a cross-tool analysis component that gathers historical data and can analyze pieces of evidence which we refer to as *indicators* (a statement of user activity from any tool in Metafora) or *landmarks* (a high-level statement of some abstract concept occurring in Metafora, indicative of accomplishment or need for remediation) that are generated by the different tools (for early steps in this approach see [13]).

Our challenge was to identify high-level student behaviors that call for intervention. From the superset of all L2L2 behaviors identified through data analysis, we select behaviors that are high-level enough to be directly relevant to L2L2 through conceptual links with the L2L2 definition, but also low-level enough to be directly mapped to certain actions within the system. Obviously, generality is a challenge, as each tool reports indicators and landmarks that are meaningful to the use of the specific tool, but not necessarily to the use of tools more generally. Therefore, we also require landmarks that can be understood in a generic sense across all tools, landmarks about which the cross-tool analysis component can reason. We have defined three broad labels for landmarks coming from the different tools that allow for cross-tool recognition and decision-making:

- *Perceived Solution*: an evaluation of an artifact produced within a tool that the students may consider a solution (but is not necessarily a solution).
- *Possible Solution*: a positive evaluation of the student's work that (based on some heuristics or criteria) is considered an acceptable solution to the given task.

- *Apparent Struggle*: some negative observation of a production process, outcome, or interaction that indicates intervention is necessary.

The cross-tool analysis component can then use these labeled landmarks and, in combination with the low-level action indicators, look for patterns across students that are indicative of L2L2 and provide opportunity for potentially fruitful intervention.

There are two distinct interventions that the automated support can send. First, a *direct message* exploits the system's interface for messages to directly present an L2L2 message (selected from the templates) to the student(s). This is a traditional form of AIED feedback, where students receive some targeted advice about their work from an automated system. This type of intervention has the advantage of directly requiring the students' attention, which can ensure students are receiving the necessary feedback. However, the direct approach has the disadvantages of being forceful and of taking control away from students.

In contrast, the second intervention method comes in the form of a *recommended message template*, a type of intervention where certain message templates in the messaging tool are highlighted in order to make clear which messages are most pertinent to the student's current situation.

We hypothesize that this recommendation intervention has multiple benefits. It has the potential to increase the students' involvement in the meta-level regulation of their own learning process, because the recommendations only hint to a student what might be most relevant, but still leave the onus on the student to engage in the L2L2 process. Additionally, a practical advantage to the recommendations is that if the AI system misjudges a situation, this will generally cause less harm. Table 1 contains examples of interventions as an outcome of analysis information shared by the tools for particular behaviors.

Table 1. Examples of mapping L2L2 behaviors to a specific pattern of indicators and landmarks that can be recognized by the cross-tool analysis component, which in turn can enact the given intervention. Examples of behaviors are related to the examples from section 2.2.

	Behavior	Indicators and Landmarks	Intervention
Distributed Lead-ership	Different members of the group should take the initia-tive to intro-duce and dis-cuss new ideas.	- One person in the group creates a new resource. - Lack of discussion (in LASAD or chat).	*Recommended Message:* "This is a new idea. We should discuss how it is relevant and how it can help us."
utual En-gage	Group should work together	- Divergence without convergence in plan-	*Recommended Message:* "Lets discuss why we have

	in a supportive and integrated way.	ning /reflection tool (Apparent struggle). - Lack of discussion (in LASAD or chat).	disagreed in LASAD, explaining first what is tricky about the task and what we are not so sure about."
Peer feedback and assessment	Group members should consider solutions offered by others and how those solutions relate to their own solutions.	-Apparent solutions from team members on separate computers -Apparent solutions not shared in LASAD, not accessed by other members	*Recommended Message:* "Lets evaluate one another's solution with respect to our task" *Direct Message:* "You should consider your solutions with respect to the task."
Group reflection	Group should re-visit and reflect upon their plan as they work	-Lack of plan revision with abundance of indicators from other tools. -Lack of attitude or Role cards	*Recommended Message:* "Let's revise the plan to show how we are going to work as a team." *Direct Message:* "You should consider how attitudes have played into your planning."

It is important to note the varied use of recommended messages vs. direct messages in the intervention column of Table 1. While each specific decision to send a direct message vs. recommendation can be debated from an instructional perspective, it is clear that certain situations may call for direct intervention because the situation is deemed as critical and the system has high confidence in its diagnosis. The difference between direct messages and recommended messages can also potentially be used as scaffolding, and faded over time. More direct messages early on can help students learn how and when these messages might be appropriate, and over time they can then be given only as recommendations, when students are expected to offer messages to one another in productive ways on their own.

Lastly, while this research is not focused on the teacher, this messaging system invites teacher participation as well, allowing them to send messages to student groups. Similarly, teachers can receive the recommendations from the system to help them quickly and easily identify the types of messages that are most likely necessary for any given group at a particular point in time. In this way, a single intervention system based on messages is acting as: 1) an intermediary for students to interact with each other, 2) a tool for teachers to interact with the students, and 3) a system for automated agents to offer intervention on varying levels of interruption.

4 Conclusion

This article presents an attempt to support social regulation in a collaborative environment known as Metafora with an explicit aim to support Learning to Learn Together (L2L2). The system, through both its design and automated support system, helps students become aware of many requirements of effectively learning with others in a group by explicitly referencing and drawing attention to the four L2L2 aspects. Since the Metafora platform and pedagogy are aimed at not only teaching domain knowledge — where approaches in AIED and ITS have demonstrated their potential — but also attempting to help students reflect on L2L2 by encouraging them to plan and regulate their own learning, we recognize that developing a 'traditional' intelligent system that sends feedback directly to students is not necessarily an adequate solution. Apart form the typical challenge of deciding when and how to provide feedback, there are conceptual challenges to ensuring this feedback encourages high-level reflection on L2L2 and that the feedback is generically available and applicable for all domains and learning scenarios.

This paper offers a new conceptualization of what an AI intervention (in the general sense) can look like: a system where fundamentally equivalent, theoretically grounded message templates can be utilized by different stakeholders (human or AI agent) according to the needs, abilities, and circumstances of the given scenario. Apart from making these message templates available for students to consider and exchange, the same basic messages can either be catered to be sent directly to students (with appropriate justification) or be recommended to students or teachers as potentially pertinent to the situation. Pilot experimentation suggests that these recommendations act not only as a practical means of helping students select from a large list of potential messages but also as a scaffold in suitable moments, to help students develop "L2L2" ways of thinking that can support them in becoming better group learners.

In future work, we intend to investigate in more detail the potential of both the availability of those messages in comparison with a less scaffold approach, and particularly the added value of the recommended messages vs. simply encouraging students to use the messaging system in general. Our hypothesis is that the sheer availability of the messages stimulates reflection and has the potential to improve awareness on L2L2. However, our previous work and initial pilots suggest that when messages are recommended based on relevance to the context, we will see even more significant behavioral changes in groups due to these messages, especially when students have ownership of the messages.

References

1. Walker, E., Rummel, N., Koedinger, K. R: Beyond explicit feedback: new directions in adaptive collaborative learning support. In Proceedings of the 9th international conference on Computer Supported Collaborative Learning - Volume 1 pp. 552-556 International Society of the Learning Sciences (2009)
2. Dragon, T., Mavrikis, M. McLaren, B.M., Harrer, A., Kynigos, C., Wegerif, R., Yang, Y.: Metafora: A web-based platform for learning to learn together in science and mathematics. IEEE Transactions on Learning Technologies. 10 Jan. 2013. IEEE Computer Society Digital Library http://doi.ieeecomputersociety.org/10.1109/TLT.2013.4 (2013).
3. Scheuer, O., McLaren, B. M.: CASE: A configurable argumentation support engine. IEEE Transactions on Learning Technologies, IEEE computer Society Digital Library. (2013)
4. Gokhale, A. A.: Collaborative Learning Enhances Critical Thinking. Journal of Technology Education, 7(1) 22-30 (1995)
5. Soller, A., Monés, A. M., Jermann, P., Mühlenbrock, M.: From Mirroring to Guiding: A Review of State of the Art Technology for Supporting Collaborative Learning. International Journal on Artificial Intelligence in Education, 15(4), 261-290 (2005)
6. Wegerif, R. Dialogic: Education for the Internet Age. Routledge (2013)
7. Dekker, R., Elshout-Mohr, M., & Wood, T.: How Children Regulate their Own Collaborative Learning. Educational Studies in Mathematics, 62(1), 57–79 (2006)
8. Dillenbourg, P., Jermann, P.: Designing integrative scripts. In Scripting computer-supported collaborative learning pp. 275–301. (2007)
9. Chaudhuri, S., Kumar, R., Howley, I., Rosé, C. P.: Engaging collaborative learners with helping agents. In Proceedings of the 2009 conference on Artificial Intelligence in Education: Building Learning Systems that Care: From Knowledge Representation to Affective Modelling pp. 365-372. IOSPress (2009)
10. Keshtkar, F., Morgan, B., Graesser, A.: Introducing an Intelligent Virtual Suggester and Assessment in Computer Educational Games. In the 11th International Conference on Intelligent Tutoring Systems (ITS-2012), Workshop on Intelligent Support for Learning in Groups 2012. Chania, Greece (2012)
11. Porayska-Pomsta, K., Mavrikis, M., Pain, H. Diagnosing and acting on student affect: the tutor's perspective. In User Modeling and User-Adapted Interaction, 18(1):125—173 (2008)
12. Mavrikis, M., Dragon, T., McLaren, B.M. Proceedings of the Workshop on Intelligent Support for Exploratory Environments 2012: Exploring, Collaborating and Learning Together at the 11th International Conference on Intelligent Tutoring Systems, ITS 2012. Chania, Greece (2012)
13. Dragon, T., McLaren, B., Mavrikis, M., Geraniou, E. Scaffolding collaborative learning opportunities: integrating microworld use and argumentation. Advances in User Modeling, 18-30 (2012)

Appendix

Table A.1. Tools used in all learning scenarios

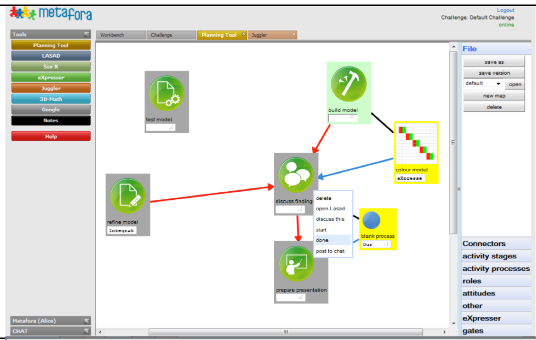
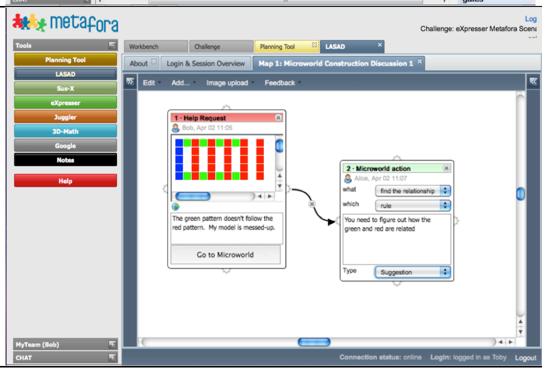
Planning/Reflection Tool: provides a visual language to support students in planning and reflecting; activities, roles, resources, task assignments, and attitudes are visualized, discussed, and reflected upon.	
Discussion Tools: provide a shared workspace for students to have in-the-moment chat, as well as structured discussions and argumentation, through a graphical argumentation tool, LASAD (see more info https://cscwlab.in.tu-clausthal.de/lasad/)	

Table A.2. Examples of message templates to be sent by students or to be recommended by the system. Note that each message also has an equivalent message with adapted language and grammar that appear as external to the group and can be used from the system as a direct message. For example, instead of "Let's look…" "Everyone should look…"

	Message Template	Comments
Distributed Leadership	Let's propose a new idea to help us explore a different direction.	Useful in a phase of brainstorming as a means of getting the team out of an impasse.
	We need to see how the new ideas are relevant and helpful to our current work.	Highlights the importance of regulatory moves during idea generation and provides an example of criteria for accepting or rejecting ideas.
	Let's look at the group planning map together.	Relevant when some students' activities seem to be diverting from the plan.
	How could we improve our plan?	Inspires specific leadership moves from members of the team. These messages promote the equal share of both work and leadership
	Let's assign tasks to	

	help us split the work equally.	(planning) from all the members of the team.
	Has everyone contributed to planning the work?	
Mutual Engagement	Has everyone done the work they said they would do?	Similar to the last two messages of the previous category, but intended to refer to engaging particularly with the discussion or work in the microworlds.
	Has everyone contributed to the discussion?	
	I/We need some help with <…>	Promotes peer help-seeking --- students are often reluctant to ask for help from peers even when stuck.
	We seem to disagree. Have we all understood each other's opinions?	Helps students step back from the "heat of the disagreement" and fosters shared understanding and by encouraging students to rethink the problem and help reach consensus and/or generate new action.
	Lets discuss our conflict starting from the causes of our confusion.	
	We seem to disagree. Lets redefine our group goals/attitudes/roles.	Defining goals/attitudes or roles involves students in a discussion about their different perspectives.
Peer feedback and assessment	We should share our models and compare them.	Sharing and comparing models promotes meaning-making with respect to the domain.
	Lets evaluate one another's solution with respect to the task.	Constructive peer assessment is an important skill but students often ignore the original task and tend to focus only on procedural rather than conceptual aspects hence this message recommends specific criteria.
	Let's explain clearly in our evaluation what is the problem	
	Let's revise our plan. Does it match our work so far?	Revising the plan at specific phases during and at the end of the collaborative process initiates reflective discussions.
	Let's use the attitude/role cards to reflect on our work so far.	Employing attitudes and roles in the plan encourages reflection on the collaborative process at the meta-level.
	Lets consider our best/worse moment as team so far.	A message often used in critical incident analysis as a way of reflecting and generating meaning out of events.

Supporting Collaborative Learning in Virtual Worlds by Intelligent Pedagogical Agents: Approach and Perspectives

Mohamed Soliman

IICM, Graz University of Technology, Graz, Austria
muhamed.soliman@gmail.com

Abstract. Intelligent pedagogical agents (IPA) are aimed to support learning in virtual worlds. Motivations for adopting IPAs in virtual worlds are to compensate for lack of human pedagogical presence, to improve student engagement, and having autonomous support. Given named challenges to realizing IPAs in virtual worlds, a proposed solution approach is to simulate IPAs with targeted scenarios with intelligent agents prior to realization. This paper discusses intelligent agent based simulation of a collaborative learning scenario that facilitates IPA support to collaborative learning in virtual world. The collaborative learning scenario is composed of multiple avatars interacting to conduct an experiment simulation in a virtual world with an IPA. The paper discusses types of support the agent will do to scaffold the interactive collaborative learning activity, for example by mediating interaction among learners and targeting learning to collaborate as well as collaborating to learn with benefits shown.

Keywords: CSCL, Intelligent Pedagogical Agents, Intelligent Agents

1 Requirements

A collaborative learning activity design is motivated by the objective to employ Intelligent Pedagogical Agents (IPAs) in virtual worlds to support learning. While there are different means to support collaborative learning in virtual worlds (Dalgarno, 2010), automated and artificially intelligent pedagogical support are still needed. Design objectives of IPAs are to provide automated and intelligent pedagogical support while improving engagement throughout interactivity. While there are different roles the IPA can do to support collaborative learning activities in a virtual world, there is the importance of focusing on interaction among learners and with a leaning object in relation to situated learning and learning by doing. Prior works (Soliman & Guetl, 2010; Soliman & Guetl, 2013) highlighted other possibilities of IPA support.

In contrary to an individual learning scenario, the IPA role has to shift towards being more of a mediator that facilitates the dialogue and interaction among the learners and the learning object in the collaborative setting. An important task of the IPA is to maintain distribution of roles, as a key component, among different learners (Hoadley, 2010). Distribution of roles in the task is assumed to be available as an input to the learning activity. The IPA is assumed to be executing a micro level script rather than a macro level to discover details of interaction as a design objective (Kollar, Fischer, & Hesse, 2006; Weinberger, 2011). Selection of the group size is determined to start with two learners agreeing to what is cited by Hoadley (2010), *"Stahl (2006) has argued that the small group level is the 'sweet spot' for studying CSCL".*

The targeted scenario is described by two avatars performing an experiment simulation with the aid of an IPA. The avatars are human controlled while the IPA is an autonomous agent. The IPA supports the learning activity with the following:

1. Provide tutorial about the experiment. In collaborative learning scenarios, the IPA will intervene only to scaffold learning after giving the opportunity to other learners to learn to collaborate.
2. Providing motivational support.
3. Answer questions. In the group learning, the IPA will rather stimulate group interaction before answering a question individually.
4. Support the collaborative activity such as "who is supposed to perform this task?"
5. Promote reflection and trans-activity (Boud, Keogh, & Walker, 1985) as important components to collaborative experiential learning.
6. Provide varying levels of support from the learner level to the group level.
7. Ensure continuation of the activity, to manage idle time behavior for example.

However, several challenges exist for implementing an IPA directly into the virtual world, Soliman and Guetl (2013). Hence, simulating the collaborative learning activity in the intelligent agent framework is useful. This is to focus on interactivity and intelligence support to the collaborative learning activity and to identify how an intelligent agent can complement the IPA functions in particular to the collaborative interaction.

2 Solution Approach

2.1 BDI-Based Collaborative Learning Scenario Simulation

The BDI agent framework of Jadex (Jadex, 2013) is adopted as a result of evaluation and selection steps (Soliman & Guetl, 2012). Inter-agent communication is used to simulate the players' interaction in the learning activity communication towards enabling its analysis and reasoning. In BDI based environments, multi-agent design involves determination of goals, plans, events (or messages), and beliefs. Goals represent static or dynamic desires the agent should pursue, plans represent intentions (as recopies of the solution) translating into actions. Beliefs represent agent knowledge about the environment and other learners and can also change dynamically according to events. A BDI based collaborative learning scenario simulation involves determination of goals, plans, and beliefs.

2.2 Settings and Design

Setting the experiment implies simulating the players (actors and artifacts) of the scenario in the virtual world. Four agents are defined: an agent representing the IPA, two agents representing the learner avatars, and an agent that simulates the intelligent object (device) behavior in the virtual world. The BDI-based agent design requires setting the beliefs, desires, and intentions of the agents:

- *The IPA agent* has beliefs about learners, the task, and the roles. The desire of the IPA is a pedagogical goal to facilitate (direct) the completion of activity. The intentions of the IPA are plans representing variations according to interactions.
- *The device agent* represents an experiment. It gives an autonomous behavior property to the object to simulate different results that can be handled in learning settings by learners or the IPA.
- *Two learner agents* are allocated. The desire of each learner agent is to accomplish the learning experiment in collaboration with another learner. Intentions adopt sequences in results to interaction. Beliefs add details of the learner knowledge about the other learner.

2.3 Interaction & Collaboration

The IPA initiates the first step to run the experiment and finds, in the role-responsibility beliefs, which learner is allocated issuing a request for the assigned learner agent to start. If the correct action is performed, it updates the assessment belief base. If the task is wrong, as observed from the device, the IPA records and triggers collaborative discussion with the other learner. The task is repeated (according to pre-set number of trials) by the same learner (if a capable learner can show the task, it can be performed by another learner). Otherwise, the IPA can give a demonstration of how the task if performed and move to the next task. The IPA will continuously monitor the interaction identifying which agent is responding. Consecutive tasks will proceed until the experiment completes. Before each step, IPA sends a message to both learners to trigger discussion on how to perform the next task. In each step, if the wrong learner responds, IPA issues an error message while recording the result into the assessment belief set. Directing messages to both learner agents serves the learning to collaborate objective. Furthermore, when the IPA recognizes long idle time, it asks both learners to discuss roles and the expected task on which action to take.

3 Concluding Remarks

The learning scenario is implemented in Jadex as a selected agent platform to avoid difficulties of actual implementation. The simulation of this scenario in the agent based environment helps to:

1. Isolate implementation difficulties in a virtual world.

2. Discover means of IPA support for collaborative scenario – how the collaboration scenarios will take place in a virtual world implementation.
3. Discover means of interaction design for the learning scenario in relation to roles.
4. Requirements from the learning object to support the learning interaction from one learner in relation to more than one learner.
5. Investigations into integrating micro-level collaborative scripts and contributing a collaborative pattern of IPA in virtual world based learning.

References

Boud, D., Keogh, R., & Walker, D. (1985). Introduction: What is Reflection in Learning. In Boud, D. et al. (Eds.), Reflection: Turning experience into learning. London: Kogan Page.

Bratman, M. Intention, Plans, and Practical Reason. (1987). Harvard University Press. Cambridge, MA, USA.

Dalgarno, B., & and Lee, M. (2010). What are the Learning Affordances of 3-D Virtual Environments? British Journal of Educational Technology. Vol 41 No 1.

Gaillet, L. (1994). An Historical Perspective on Collaborative Learning. Journal of Advanced Composition, 14 (1).

Hoadley, C. (2010). Roles, Design, and the Nature of CSCL. Journal of Computers in Human Behavior. Vol 26, Issue 4 (pp. 551-555).

Jadex, (2013). http://jadex-agents.informatik.uni-hamburg.de, Accessed May 31, 2013.

Jaques, P., Andrade, A., Jung, J., Bordini, R., & Vicari, R. (2002). Using Pedagogical Agents to Support Collaborative Distance Learning. CSCL'02, Boulder, Colorado.

Kobbe, L., Weinberger, A., Dillenbourg, P., Harrer, A., Hämäläinen, R., & Fischer, F. (2007). Specifying Computer-supported Collaboration Scripts. International Journal of Computer-SupportedCollaborative Learning, 2(23), 211–24.

Kollar, I., Fischer, F. & Hesse, F. W. (2006). Collaboration Scripts – a Conceptual Analysis. Educational Psychology Review, 18(2), 159–85.

Schmeil, A., & Eppler, M. (2009). Knowledge Sharing and Collaborative Learning in Second Life: A Classification of Virtual 3D Group Interaction Scripts. International Journal of Universal Computer Science (JUCS), 15(1).

Soliman, M., & Guetl, C. (2013). Implementing Intelligent Pedagogical Agents in Virtual Worlds: Tutoring Natural Science Experiments in Open Wonderland. The IEEE Global Education Conference, IEEE EDUCON 2013, March 12-15 2013, Berlin, Germany.

Soliman, M., & Guetl, C. (2012). Experiences with BDI-based Design and Implementation of Intelligent Pedagogical Agents. International Conference on Interactive Computer-Aided Learning, ICL2012, Sept. 2012, Villach, Austria.

Soliman, M., & Guetl, C. (2010). Review and Perspectives on Intelligent Multi-agent Systems' Support for Group Learning. World Conference on Educational Multimedia, Hypermedia & Telecommunications ED-MEDIA 2010, June 2010, Toronto, Canada.

Stahl, G., Koshmann, T., Suthers, D. (2006). Computer Supported Collaborative Learning: a Historical Perspective. In Sawyer (Ed.), Cambridge handbook of the learning sciences, Cambridge University Press.

Vygotsky, L. (1978). Mind in society. The development of higher psychological processes. Cambridge: Harvard University Press.

Weinberger, A. (2011). Principles of Transactive Computer-Supported Collaboration Scripts. Nordic Journal of Digital Literacy. No. 03.

Towards semantic descriptions of collaboration indicators to support collaboration models transferability

Jesús L. Lobo[1], Olga C. Santos[2] and Jesus G. Boticario[2]

[1] Technology & Society, Tecnalia Research & Innovation, Basque Country, Spain
[2] aDeNu Research Group. Artificial Intelligence Dept. Computer Science Sch. UNED. Spain
[1]jesus.lopez@tecnalia.com, [2]{ocsantos,jgb}@dia.uned.es

Keywords: Collaborative learning, Transferable collaboration model, Collaborative indicators, Collaboration analysis

1 Research

Students around the world are currently taking advantage of e-learning platforms to support their learning, and one of the most important features in some of these platforms is their support for collaborative learning. In this context, a collaboration analysis is necessary to ascertain whether collaboration takes place. Having this in mind, data mining techniques are often used to identify student collaboration indicators based on their forum interactions (see relevant literature elsewhere).

The Collaborative Logical Framework (CLF) system, based on an approach used by international Cooperation Agencies, sets guidelines to promote participation in CSCL [1]. It is fully integrated into dotLRN/OpenACS as one of its packages and consists of making the students work consecutively in three ways: 1) solving tasks individually 2) working in cooperation with their colleagues' to improve own solutions, and 3) working all together to reach an agreement for the joint solution. Moreover, the system gathers the students' performance to infer how they work in the course. By means of a varied number of metrics, derived from the analysis of forum interactions, the system provides their behavior related to the collective task. In particular, these metrics focus on ratings given to their colleagues' contributions, on the revised versions they create of their solutions after the colleagues feedback received, and studying the actions they carry out before and after a specific interaction. This information helps the student and the tutor to monitor the tasks, and on the other it is used to get collaborative indicators, which define the learner's reputation.

Domain-independent statistical indicators of students' interactions in forums (conversations started, messages sent, and replies to student interactions) were identified elsewhere by mining non-scripted interactions in dotLRN and evaluated the benefits of their awareness by students [2]. In this context, the objective of this work is to enrich student's meta-cognitive support in the CLF by adding these automatically inferred and validated indicators (focused on initiative, activity and regularity, and perceived reputation) using the CLF metrics to express them.

If possible, our intention is to use available standards and specifications to semantically model the indicators and support transferability of collaboration models among different systems.

Besides well-known benefits of collaboration awareness in motivating students' collaboration, indicators inferred can be also used to provide adaptive features to the e-learning system. Thus, depending on the student collaboration profile and behavior, the system can react accordingly by providing individual suggestions. The goal here is to identify recommendation opportunities that guide the student to perform specific actions in order to help on the task, encourage participation and improve team work.

2 Suggested Topics for Discussion

- Descriptions of collaboration indicators modeling in terms of available standards to support transferability of collaboration models among systems.
- Elicitation of recommendation opportunities to manage and guide collaboration.

3 Biography

Mr. Jesús L. Lobo has an MSc in Computer Engineering (Deusto University, Spain, 2003) and he is working at Tecnalia Research & Innovation as Projects Responsible and ICT Consultant. His work is mainly focused on key activities such as e-skills, technology for learning, ICT certification processes, and ICT and lifelong learning activities.

Dr. Olga C. Santos is aDeNu's R&D Technical Manager and has contributed to 14 projects and over 150 papers and 50 scientific committees researching on affective inclusive personalized adaptive navigation support in ubiquitous standards-based social online learning environments.

Dr. Jesus G. Boticario is professor at the Computer Science School and has held several positions at UNED. He has published over 200 research articles and participated in 24 R&D funded projects. Head of aDeNu research group, scientific coordinator in European and National projects and co-chairs of workshop series on User Modeling and Accessibility.

Relevant Publications

1. Santos, O.C., Rodríguez, A., Gaudioso, E., Boticario, J.G.: Helping the tutor to manage a collaborative task in a web-based learning environment. In: AIED2003 Supplementary Proceedings, 153-162 (2003)
2. Anaya, A.R., Boticario, J.G.: Content-free collaborative learning modeling using data mining. In: User Modeling and User-Adapted Interaction 21, 181-216 (2011)

Scripting at the tabletop to improve collaboration

Andrew Clayphan

School of Information Technologues, The University of Sydney, NSW, 2006, Australia
andrew.clayphan@sydney.edu.au

Keywords: Tabletops, Interactive Surfaces, CSCL, Scripting

1 Research

Interactive collaborative tabletops are promising devices that can help collocated people collaborate because they augment natural round-table discussions with a shared digital space that offers equal opportunities of actions and access to resources available. We propose collaborative scripts for enhancing tabletop collaboration in the form of: guidance and structure; advice on how to do the task; and control over constraints afforded by the tabletop.

After studying the ways people have used tabletop interfaces, we concluded that it is valuable to define scripts that will help people collaborate more effectively in co-located, technology-enhanced scenarios [3]. Different from scripts investigated so far, our work allows learners to negotiate over the scripts – initially explored in the domains of brainstorming, concept mapping, and collaborative poster creation.

Brainstorming – a technique to encourage creativity in small groups. Our method separates the technique into three stages: idea generation; idea organisation and reflection [1]. Each stage is scripted through the use of negotiation elements that alter a stage. The system presents a choice between users leading negotiation or a facilitator making choices, for example: whether to enable touch input; whether to colour ideas (to show authorship); etc.

Concept Mapping – a technique to help learners represent knowledge about a given topic in a graphical format, making use of meaningful propositions to link concepts in a domain of interest. Building a concept map at the tabletop can help students visualise different perspectives of the same topic and trigger discussions towards agreement on main ideas that describe the knowledge domain [4]. Collaborative scripts are set to drive groups of students to produce better quality concept maps, for example: the layout of concepts according to different theoretical principles.

Collaborative Poster Creation – designed for small groups to build a joint artefact from personal collections [2], consisting of an individual collection stage, and then collaborative stages of sharing and building. The collaborative stages have potential for scripting, for example: enforcing viewing of content – before being permitted to advance in the task.

Figure 1. Examples of tabletop applications used for exploring scripting.

Each activity, presents design issues to consider when formulating a set of guidelines to consider for scripting at the tabletop. These are: (1) People have different expectations and knowledge of the task at hand. (2) Voting/negotiation mechanisms – the way a group resolves issues. (3) The need for sound default settings. (4) Identifying group collaboration and how to show this to learners. (5) Whether the main task was executed as expected, and the role scripting had towards this.

We propose a set of guidelines: (1) Regulate learning activities [6] – keep "*activities of learners coordinated and guided according to particular rules, implemented via respective tools in the learning environment*" [5]. (2) Foster collaboration – organise the activity and the script to promote collaboration. (3) Facilitate egalitarian participation. (4) Define level of user control. (5) Foster awareness – develop an understanding of other participant actions. (6) Adjust the script based on information from the system and the users. (7) Use Tabletop Affordances – take advantage of the constraints introduced by the tabletop, such as: face to face discussion; and methods to exploit the hardware.

2 Suggested Topics for Discussion

- Whether script approaches at the tabletop should be system or role based or both?
- The representation of open learner models to aid in the scripting process?
- The appropriate level of feedback for learners? OLM's?
- Methods to help determine if a script is needed?

3 Biography

Andrew Clayphan is a Ph.D. student at the CHAI Research Group at Sydney University, Australia. He holds degrees in Software Engineering (University Medal, Honours Class 1) and Finance from the University of New South Wales, Australia.

Relevant Publications

[1] A. Clayphan, J. Kay, and A. Weinberger. ScriptStorm: scripting to enhance tabletop brainstorming. In Personal and Ubiquitous Computing, (to appear), 2013.

[2] A. Collins, A. Clayphan, J. Kay, and J. Horder. My museum tour: Collaborative poster creation during school museum visits. In EIST2012.

[3] L. Kobbe, A. Weinberger, P. Dillenbourg, A. Harrer, R. Hämäläinen, P. Häkkinen, and F. Fischer. Specifying computer-supported collaboration scripts. In IJCSCL 2(2):211–224, 2007.

[4] R. Martinez, J. Kay, and K. Yacef. Collaborative concept mapping at the tabletop. In ITS2010, pages 207–210, 2010.

[5] A. Meier, H. Spada, and N. Rummel. A rating scheme for assessing the quality of computer-supported collaboration processes. In IJCSCL, 2(1):63–86, 2007.

[6] A. Weinberger. Principles of Transactive Computer-Supported Collaboration Scripts. Nordic Journal of Digital Literacy, 03:189–202, 2011.