Capturing and Analysing the Processes and Patterns of Learning in Collaborative Learning Environments

Kate Thompson, Shannon Kennedy-Clark, Lina Markauskaite, Centre for Computer Supported Learning and Cognition, University of Sydney, Sydney, Australia,
Email: kate.thompson@sydney.edu.au, shannon.kennedy-clark@sydney.edu.au, lina.markauskaite@sydney.edu.au
Vilaythong Southavilay, School of Information Technologies, University of Sydney, Sydney, Australia, vstoto@it.usyd.edu.au

Abstract: This paper describes our methodological experiences capturing and analyzing student learning processes and patterns in three cases. Agent-based models and a virtual world were used for learning. Several specific features tie these cases together and distinguish our analysis from other studies in the CSCL domain. First, students interacted in real-time for relatively short periods. Second, they interacted with each other and with interactive software tools that dynamically 'shaped,' and were shaped by, their learning process. Our work builds upon and integrates process analytic approaches of dynamically captured video and computer screen activity and automatic e-learning process analysis techniques. The first two cases identify areas in which analysis 'by hand' of small amounts of data has produced findings of initial interest. The third case discusses the use of an automatic pattern discovery technique based on Hidden Markov Models to begin to apply these methods to larger data sets in CSCL environments.

Introduction

Research on computer-supported collaborative learning (CSCL) has made significant progress in capturing and analysing student communication and decision-making processes in collaborative learning environments. In particular, this can be said about the number of studies that have explored how students navigate, communicate and collaborate in asynchronous learning management systems (Dringus & Ellis, 2005; Macfadyen & Dawson, 2010). Similarly, significant progress has been made in capturing and exploring human-computer interaction with learning software. These methods range from capturing and analysing software log files, to video observations, screen recordings, and eye-tracking traces (Cox, 2007; Derry, et al., 2010). The main enabler of this type of research is a variety of digital traces of students' interaction within and with software, captured in digital media. As the NSF's Taskforce on Cyberlearning report indicates, these learning traces could "aid researchers in developing a more complete and accurate scientific understanding of what makes learning most productive and enjoyable" (Borgman, et al., 2008, p. 26).

The field of CSCL encompasses a variety of scales, methods of collaborating, and media (Dillenbourg, 1999). Thus, CSCL researchers inevitably have to deal with very diverse (often multimodal) data and, as Strijbos and Fischer (2007) argue, such research often demands the integration of different analytical techniques. The use of Hidden Markov Models (HMMs) (Rabiner, 1989) is one technique that can be used to discover behavior patterns of student collaboration and interaction. Based on a sequence of activities, which can be captured in students' trace data or log files, the HMMs can extract the states that students go through as well as the transitioning probabilities among these states. A student's problem-solving behaviors and patterns can then be derived by analyzing the activities associated with the states and the state transitions.

The advantages of studying the processes of interaction in CSCL research are regularly discussed and the links with learning outcomes often made (for example, Cox, 2007). Less common is a simultaneous analysis of the process of student discussion with each other and dynamic interaction with the *software tool* used for learning (Thompson & Reimann, 2010). The integration of these two areas provides the focus for this paper.

Our aim is to illustrate how student interaction analyses using automatically captured data could be further extended. We will present two earlier studies that have used computer screen capture to analyze collaborative decision making and strategies for interaction with a software tool. We conclude with a recent pilot study in which student interaction patterns with software have been extracted automatically using HMM.

Case Study 1 - Collaborative Decision Making

This case study focuses on the use of screen recording software (*Camtasia)* in a virtual world (*Virtual Singapura*), which allowed the recording of both a video of students' use of the tools and the interaction between students in a dyad. We recorded the in world actions of four post graduates and eight undergraduates. *Virtual Singapura* is a virtual world that is based on disease epidemics in 19th Century Singapore. The participants were provided with a paper-based activity that focused on reducing cholera in the city. The participants completed their in-world activity in pairs. The activity took approximately 40 minutes to complete.

The recordings provided three sources of information: audio, video and screen shots. The audio transcriptions were coded according to a modified version of the Decision Function Coding System (DFCS) (Poole & Holmes, 1995). The DFCS has seven main categories: 1) problem definition; 2) orientation; 3) solution development; 4) non-task; 5) simple agreement; 6) simple disagreement; and 7) implementation. Category Solution Development (3) has five subcategories: 3a solution analysis, 3b solution suggestions, 3c solution elaboration, 3d solution evaluation, 3e solution confirmation.

Camtasia captured pertinent information regarding the aspect of the environment that the learners were focusing on (Mazur & Lio, 2004). Many processes were represented by an action rather than a verbal interaction. Activity sequences were recorded and the process sequences mapped. Figures 1 and 2 demonstrate the main processes that pairs engaged in, in order to arrive at problem solutions.

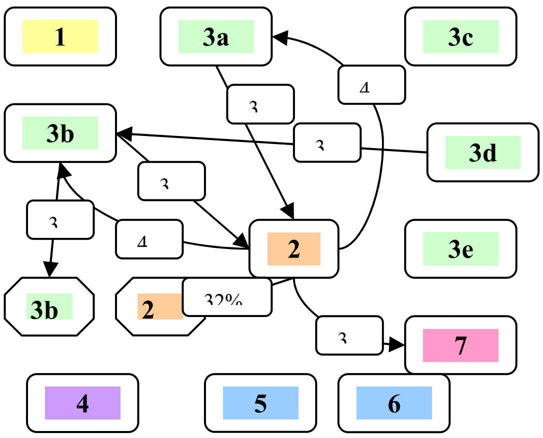

Figure 1. Group 1 Decision Making Processes.

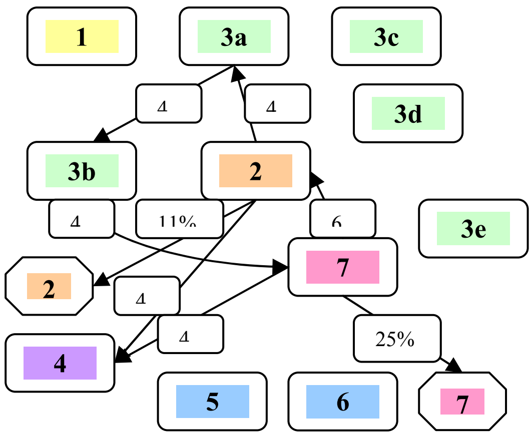

Figure 2. Group 2 Decision Making Processes.

The results of the analysis indicated that groups that were successful at arriving at a conclusion were more likely to use orientation processes (code 2) as a means of directing the group. The highest proportion of events tended to be an orientation event followed by a subsequent orientation event. The groups that did not arrive at a successful conclusion (e.g. Group 2 see Figure 2) were more likely to have non task events, such as talking about another subject. More detail on these findings can be found in more detail in (Kennedy-Clark, Thompson, & Richards, accepted). The opportunity to relate the synchronous collaboration data to the information about the interaction with the virtual world provided insights into the design and scaffolding required in this inquiry-learning task. However, coordinated analysis of data was time-consuming.

Case Study 2 - Collaborative Use of Agent-based Models

In this case, the main focus is the identification of strategies that students use to interrogate agent-based models. As with Study 1, the recording of the data and the identification was performed manually. Video screen shots were collected from two dyads of year 9 school students and coded with respect to times, activities and screens. This study used an agent-based model built in NetLogo (2010) which focused on the impact of visitors in a National Park (Thompson & Reimann, 2010). Information on the use of the model, such as the screen accessed (information or experiment), the number of times each of the three variables was changed, the model was run and the total activity (including these and other activities not discussed here) were also recorded, as was time off-task. The patterns were classified according to Levy and Wilensky's (2005) strategies (see Table 1).

Table 1: Strategies of exploration in agent-based environment found in Levy and Wilensky's (2005) study.

	Strategy		
Name	Straight to the point	Homing in	Oscillating
Overall observation time	Lower	Lower	Higher
Observation time per run	Higher	Lower	Lower
Time between actions	Higher	Lower	Lower
Runs	Lower	Higher	Medium

The time observing the model was taken as the time spent on the experiment screen. The time spent observing the model in each setting was calculated by dividing the total time spent observing the model by number of times 'go' was selected. The time spent off-task and spent reading the text/instructions were added as a result of a pilot study. The number of runs was equal to the number of times 'go' was selected. Time per action was calculated by dividing the time observing the model by the number of changes made, and the number of changes made was equal to the total activity.

Table 2: Patterns of use of the two dyads.

Strategy	Group 1	Group 2
Observation		
Time observing the model	16:20 (Higher)	14:01 (Medium)
Time observing the model in each setting	2:43 (Higher)	2:20 (Medium)
Time spent off task	0:00	2:32
Time spent reading text / instructions	3:40	3:23
Explorativeness		
Number of runs	6 (Medium)	6 (Medium)
Action		
Time per action	0:43 (Medium)	0:32 (Medium)
Number of changes made	23 (Medium)	26 (Medium)
Pattern	Oscillating	Oscillating

Table 2 presents results of two dyads. In these two groups only the oscillating (the model oscillates between two regimes, back and forth between high and low values) strategy was identified (other strategies are reported in (Thompson & Reimann, 2010)). The strategies used by students to change the three variables were also determined using graphs of the changes in addition to the parameters outlined above. As part of this study, preliminary work was carried out that found qualitative relationships between the types of observations recorded by students in the answers to post-test questions and the oscillating strategy that was used to interrogate the model. An advantage of using agent-based models is that students are able to identify links between levels of a system (Stieff & Wilensky, 2003) - the ability to examine *how* students use these models helps to discover learning gains. This is another instance where the coordinated analysis and collection of this data was time-consuming. In order to find useful patterns and relationships, a larger sample size is required. The automated collection of log-files that record key-strokes can aid this, but the discovery of patterns of use is still nascent. The following case reports an initial investigation of students' problem-solving strategies using HMM.

Case Study 3 - Exploring Students' Problem-solving Strategies from Log Files

In this section, we demonstrate automated process analysis techniques of students' problem-solving behaviours. A pilot study was conducted to analyse sequences of students' activities in solving scientific problems. Six students were asked to use an agent-based model built in NetLogo, similar to the models described above, however the emphasis was on learning physics (Coulomb's law) (Sengupta & Wilensky, 2005).

Each student had two screens adjacent to each other: an information screen and an experiment screen. On the information screen, students entered the results of their experiments and the answers to the activities. On the experiment screen, students explored the strength of the attractive force and the distance (movement) between two charges (q1 and q2). Students were able to change several parameters: the value of the charge, the fade-rate value, the permittivity value, and the speed of the simulation. After setting up the two charges, students could run the simulation by pressing the GO button and then moving q2 around in the model using the mouse.

The video screen shots, recorded with *Camtasia,* were manually coded into log files of students' activities. Each record consisted of the student action, the timestamp, and the screen on which the corresponding action happened. In the next step students' interactions with the models were re-coded as one of 14 activities: SE (set-up); GO (go); CC (change charge); CP (change permittivity); CR (change fade-rate); CS (change speed); CL (click on screen); MQ (move q2); SQ (stop move q2); TA (type answer); TT (type in table); DT (delete table); TE (inactivity on information (text) screen); and MO (inactivity on experiment (model) screen). We intentionally included inactive events (pauses) on the information and experiment screens as they represented events when students observed the models without changing any configuration or setup (experiment screen) and read the text without typing anything (information screen).

The generated sequences were used to derive student activity patterns. The HMM generating algorithm described in Jeong et al. (2010) was applied. Two derived problem-solving process patterns - a model-oriented (M) and text-oriented (T) - are depicted in Figures 3 and 4. Each HMM is made up of a set of states: the student activity patterns (the output probability) associated with each state and the transition probabilities between states. For example, the M student in the B state focused on the Netlogo model screen 32% of the time, and explored the model by moving the q2 charge around and stopping the charge after moving it 68% of the time (34% each for moving and stopping activities). The probability associated with a link between the two states indicates the likelihood of students transitioning from one state to another. For instance, the M model predicts that in the B state: after the student explored the model by focusing on the model screen, moving the charge and stopping the exploration, the student then entered the results and answers with a likelihood of 28%, restarted the model execution with a likelihood of 2%, changed the parameters with a likelihood of 2% or remained in the same state with a likelihood of 67% (i.e. the student continued exploring the model with the same setting).

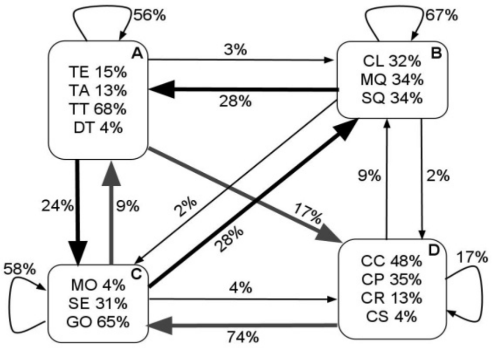

Figure 3. HMM of the Model-oriented Student (M).

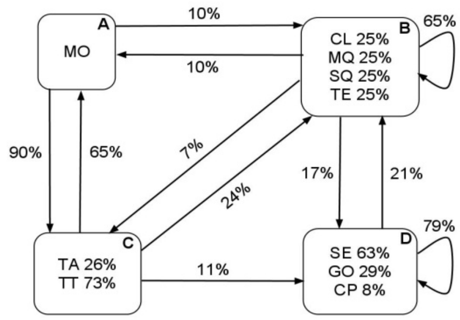

Figure 4. HMM of the Text-oriented Student (T).

We investigated the two HMMs, M and T, to gain insight into how both students solved problems. Although both models have four states, they have different structures: activities associated with each state and the state transition behaviours. We labelled the states of both models according to the activities associated with each state. The M model consisted of the following states: text (A), exploration (B), experiment setup/execution (C), and parameter configuration (D). The T model had one state (A) consisting of only one activity, MO, which represented an inactive event (pause) on the experiment screen. It also combined parameter configuration and experiment setup/execution states into one state because it had only one configuration activity, CP, which was in the same state (D) as GO and SE. Overall, the T model had the following states: inactivity/observation (A), exploration (B), text (C), and experiment setup/configuration/execution (D).

Figure 3 shows the two major paths that the M student was likely to go through in solving the problems: D-C-A-D and C-B-A-C. In the D-C-A-D path, the student started the activities with parameter configuration (state D) and then ran an experiment (C). The student then entered the results in the table (A) before returning to parameter configuration (D) to set up and run another experiment. This D-C-A-D path shows the activity pattern of parameter configuration and experiment execution. The C-B-A-C path represents the pattern of exploratory activities. When running the experiment (state C), instead of entering data into the table provided, this student explored the distance and the forces between the charges (B) before noting down the results and answers (A). The student was then likely to set up and restart the experiment in order to explore more. It was interesting to note the high transition probabilities of the loops in states B (67%), C (58%) and A (56%), compared to D (17%). When the student arrived at state B, they spent some time engaging in exploratory activities before moving to the next state. The student also spent a considerable amount of time setting up the charges before the experiment (C) and entering information and answers (A).

In contrast, there was no obvious path that the T student followed while attempting to solve the problems. Figure 4 indicates that the transition probabilities of the loops of states D and B were quite high at 79% and 65%, respectively. Although the T student spent a lot of time setting up and running the experiments (D) and exploring the forces and distances between charges (B), they often used the same parameter configurations. In addition, states B and D were coupled closely to each other: when the student was at D they would either remain in that state or transfer to B and after some time the student was more likely to transfer back to D. Although the student did read the questions in the information screen (i.e. TE activity in state B), this student did not note down the results in the table. After conducting an experiment (B and D), this student was likely to move to the table and answer the questions (C) by either passing through state A or transferring directly from D to C. However, when entering information into the table, the student was likely to stare at the experiment screen without changing the model, and then return to the table. This was indicated by the high transition probabilities from A to C (90%) and from C to A (65%).

The M and T models are two of the common behavior patterns found in this initial analysis. Further identification of the strategies described in Study 2, and determination of the relationship between these and learning outcomes are needed. In future work, we plan to collect more detailed log file data automatically which will allow us to build more elaborated patterns of student model use and collaboration.

Discussion and Conclusions

We discussed three case studies that demonstrate several approaches for the analysis of student interactions with software and between collaborating students. We showed the possibilities of integrating screen capture data with the automatic discovery of processes patterns. In study 1, screen recordings allowed the simultaneous analysis of interactions both within the dyad and with the MUVE. We were able to code recordings using the DFCS and analyze student collaboration and decision making in the MUVE. The integration of verbal and screen capture data in this case resulted in a better understanding of the decision-making of the dyads than if only the discussions were analysed. In study 2, the analysis of screen capture data allowed important relationships between student strategies of interacting with agent-based models and particular learning outcomes to be established. Study 3 demonstrated the way in which we can use the automatic discovery of processes patterns, and what such patterns could indicate about student learning strategies. To this end, we demonstrated several of the possibilities of the use of HMM to extract students' problem-solving behaviors. All three cases together illustrated that such process analyses may allow us to understand how students *learn* in CSCL environments and what kind of learning processes various combinations of particular collaborative pedagogies and computer supported learning environments can afford.

References

Borgman, C. L., Abelson, H., Dirks, L., Johnson, R., Koedinger, K. R., Linn, M. C., et al. (2008). Fostering Learning in the Networked World: The Cyberlearning Opportunity and Challenge, A 21st Century Agenda for the National Science Foundation. Arlington: NSF Task Force on Cyberlearning.

Cox, R. (2007). Technology-enhanced research: educational ICT systems as research instruments. Technology, Pedagogy and Education, 16(3), 337 - 356.

Derry, S. J., Pea, R. D., Barron, B., Engle, R. A., Erickson, F., Goldman, R., et al. (2010). Conducting Video Research in the Learning Sciences: Guidance on Selection, Analysis, Technology, and Ethics. Journal of the Learning Sciences, 19(1), 3 - 53.

Dillenbourg, P. (1999). What do you mean by collaborative learning? In P. Dillenbourg (Ed.), Collaborative-learning: Cognitive and Computational Approaches (pp. 1-19). Oxford: Elsevier.

Dringus, L. P., & Ellis, T. (2005). Using data mining as a strategy for assessing asynchronous discussion forums. Computers & education, 45(1), 141-160.

Jeong, H., Biswas, G., Johnson, J., & Howard, L. (2010). Analysis of productive learning behaviors in a structured inquiry cycle using hidden markov models. Paper presented at the International Conference on Educational Data Mining.

Kennedy-Clark, S., Thompson, K., & Richards, D. (accepted). Collaborative problem solving processes in a scenario-based multi-user environment. Paper presented at CSCL 2011, 9th International Conference on Computer-Supported Collaborative Learning.

Levy, S. T., & Wilensky, U. (2005). An analysis of student' patterns of exploration with NetLogo models embedded in the Connected Chemistry environment. Unpublished draft to be presented at the 2005 annual meeting of the American Education Research Association, Montreal, Canada. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL.

Macfadyen, L. P., & Dawson, S. (2010). Mining LMS data to develop an "early warning system" for educators: A proof of concept. Computers & Education, 54(2), 588-599.

Mazur, J., & Lio, C. (2004). Learner Articulation in an Immersive Visualization Environment. Paper presented at the Conference on Human Factors in Computing Systems, Vienna, Austria.

NetLogo. (2010). NetLogo. 2010, from http://ccl.northwestern.edu/netlogo/

Poole, M. S., & Holmes, M. E. (1995). Decision Development in Computer/Assisted Group Decision Making. Human Communication Research, 22(1), 90-127.

Rabiner, L. (1989). A tutorial on hidden markov models and selected application in speech recognition. Paper presented at the IEEE.

Sengupta, P., & Wilensky, U. (2005). NetLogo Electrostatics model. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL.

Stieff, M., & Wilensky, U. (2003). Connected chemisty - Incorporating interactive simulations into the chemistry classroom. Journal of Science Education and Technology, 12(5), 285-302.

Strijbos, J.-W., & Fischer, F. (2007). Methodological challenges for collaborative learning research. Learning and Instruction, 17(4), 389-393.

Thompson, K., & Reimann, P. (2010). Patterns of use of an agent-based model and a system dynamics model: The application of patterns of use and the impacts on learning outcomes. . Computers and Education, 54(2), 392-403.

Acknowledgements

Case Study 3 is based on the pilot study for the Australian Research Council grant LP100100594. Lina Markauskaite acknowledges the contributions of the other chief investigators, Michael Jacobson and Chun Hu.