A Tool for Integrating Log and Video Data for Exploratory Analysis and Model Generation

Victor Girotto, Elissa Thomas, Cecil Lozano, Kasia Muldner,
Winslow Burleson, and Erin Walker

Computing, Informatics & Decision Systems Engineering, Arizona State University
{victor.girotto,eethomas,cecil.lozano,katarzyna.muldner,
winslow.burleson,erin.walker}@asu.edu

Abstract. Analysis of students' log data to understand their process as they solve problems is an essential part of educational technology research. Models of correct and buggy student behavior can be generated from this log data and used as a basis for intelligent feedback. Another important technique for understanding problem-solving process is video protocol analysis, but historically, this has not been well integrated with log data. In this paper, we describe a tool to 1) facilitate the annotation of log data with information from video data, and 2) automatically generate models of student problem-solving process that include both video and log data. We demonstrate the utility of the tool with analysis of student use of a teachable robot system for geometry.

Keywords: log analysis tool, cognitive modeling, intelligent tutor.

1 Introduction

With the advent of the web and the ubiquity of computing, log data from educational systems is dramatically increasing [1]. Analysis of log data that contains information about interactions with these systems helps researchers create different expert and novice models of student behavior [2, 3, 4]. These models allow intelligent systems to adapt to the learner's knowledge, ability and needs [5] by tracking interactions and making inferences about what students know or need feedback on [6].

Some forms of interaction are better captured in video, such as gestures, discussion with teachers, or even the use of non-digital materials. Analysis of video can be very time-consuming [7], and therefore a number of different tools have been developed, mainly to support annotation. AVnnotator [8] facilitates the addition of contextual information with a variety of lenses that allows users to document different categories of information to any given scene in a video file. Other tools allow exporting annotations so that researchers can manually integrate these with other analysis resources [9]. However, none of these tools integrate video and system logs.

This paper presents a tool that facilitates integrated analysis of data in these two formats. Our approach leverages behavior graphs, first proposed by Newel [10]. McLaren et al., in an approach called Bootstrapping Novice Data (BND), demonstrated how log data could be used to automatically generate behavior graphs [11]. Like McLaren,

S. Trausan-Matu et al. (Eds.): ITS 2014, LNCS 8474, pp. 69–74, 2014.

our tool supports automatic generation of behavior graphs, but extends it by supporting the integration of video code data. This enables users to annotate logs with additional information obtained through video analysis and to see their annotations automatically represented on the behavior graphs. This approach makes two contributions: 1) We incorporate video data in the automatic graph generation, and 2) We provide a visualization that makes it possible to easily explore relevant expert and novice models from the graph characteristics.

In the remainder of this paper, we describe our tool and then present a proof of concept analysis of its efficacy using data collected from student use of our *Tangible Activities for Geometry* (TAG) system. TAG is an embodied environment that supports middle school students during learning of geometry in a digitally-augmented physical space [12]. It uses a teachable agent paradigm, where students are told they will tutor a robot named Quinn to solve problems such as “Plot the point (3,1)”. They can do so by moving within the physical space to give Quinn commands such as Move *N* units, or Turn *M* degrees. When students believe they have solved the problem they can check the correctness of their solution and receive feedback from TAG. In theory, students benefit from using TAG by encoding the relevant geometry concepts in an embodied way, and by making their reasoning explicit while instructing Quinn on how to solve the problems. Through the models generated by our tool, we can explore the strategies, misconceptions, preferences and influence of embodiment on student performance. Ultimately, this information can help the system in tailoring the feedback given to students during problem-solving.

2 The Analysis Tool

In order to understand how a system like TAG can provide better adaptive guidance to students, there is a need to identify student strategies and misconceptions, and how these interact with students’ embodied behaviors. The analysis tool that we developed generates, for both aggregated and individual student log data, a behavior graph (defined below). The tool syncs log data with video data; a researcher can annotate the log data with video information as he or she views a student’s video. The behavior graph is updated as annotations are made, giving visual insight on the relationship between video data such as gestures and actions performed. Fig. 1 shows the main interface of the application, comprised of two main sections: log/video information (items A and B) and the behavior graphs (items C and D).

The behavior graphs consist of *states* (location and orientation of the teachable agent in the coordinate plane; represented by circles) and *transitions* between states, corresponding to actions (e.g., *moving from (0,0) to (1,0)*; represented by lines). Characteristics of the underlying data are visually encoded as follows. The node size and transition thickness are proportional to how many students passed through them, with larger nodes or thicker transitions indicating more students. Color is also used: a blue node indicates the starting state; green and red indicate correct and incorrect states, respectively; and color intensity characterizes the number of students who have checked their solution at that state, resulting in a white node if no student checked the solution at that particular state. The text inside of the nodes shows the state that it represents, using the format *x*|*y*|*orientation* (e.g. 2 | -1 | 90).

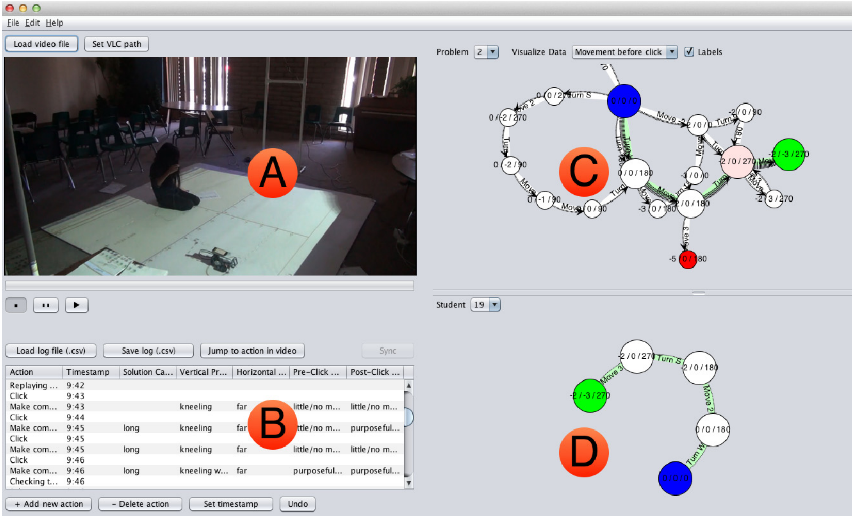

Fig. 1. The tool's main interface. (A) Video viewport. (B) Log table. (C) Aggregate behavior graph. (D) An individual student behavior graph.

The tool includes two graphs: the *aggregate graph* shows information from all students for any given problem (Fig. 1C) and the *individual graph* shows information from a single student (Fig. 1D). Given the high amount of data that needs to be represented in the graph, users can interact with the graph in several ways. It is possible to change problems and students, enable or disable labels, pan, zoom, move nodes around, and switch between the visualization of different annotations. Clicking on a node or transition displays detailed information, such as which students passed through it or how many checked for correctness.

The final aspect of this tool is its support of a seamless two-way navigation between the video and logs, making it easier to sync actions and to enable annotation of the log file based on video information. The tool automatically highlights the current action in the imported log file as the user plays the video. Alternatively, the user can move through the log file and the video will automatically sync to the log location. Users can annotate log entries using free-form text through the table seen in Fig. 1B. Each annotation is associated to its respective edge on the graph and receives a weight. The graph then encodes these weights visually by coloring its edges, using a darker green to denote a higher occurrence of this annotation, and a lighter one to denote a lower occurrence.

3 Using the Tool for Analysis: Proof of Concept

We used this tool to analyze data from a prior study [13], with the goal of better understanding student behavior in order to guide future developments of the system.

In that study, 19 subjects (8 female, 11 male) spent 45 minutes teaching Quinn how to solve point-plotting problems. Interactions were recorded both in the system's logs and in video, which were loaded into the tool. We will now describe the four exploratory analyses done using the tool.

Metacognitive Strategies. Through visual inspection of the graphs, we derived a set of metacognitive strategies that students used to solve problems. The strategies identified were: 1) W*andering*: the student follows a long path that does not lead to the solution (used by 3 participants). 2) C*hecking and resetting*: the student follows a path, checks it, and if incorrect, restarts the problem and tries a different approach (used by 11 participants). 3) C*onstant checking*: the student checks the answer after most actions (used by 3 participants). 4) *Intelligent novice*: the student takes a slightly long path to the correct solution (used by 13 participants). 5) *Expert*: the student moves directly towards the correct solution (used by 12 students). This information can aid the system in intervening positively to improve student performance.

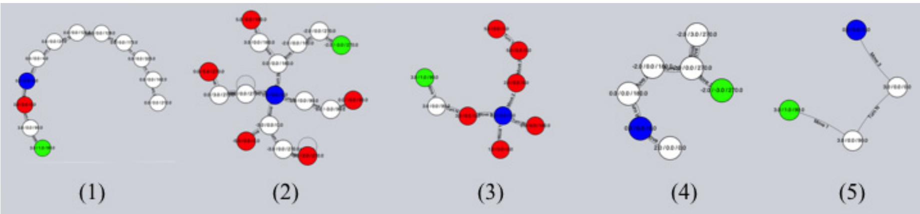

Fig. 2. Visualization of the metacognitive strategies taken by students while solving problems in the TAG system. (1) Wandering (2) Checking and resetting (3) Constant checking (4) Intelligent novice (5) Expert.

Bug Taxonomy. We also used the behavior graph to identify the nodes where students submitted an incorrect response, and classified their misconceptions. We identified several common misconceptions across students. Some examples are: 1) *Sum coordinates*, student summed the two numbers in the coordinate and move that amount in one arbitrary axis. 2) *Switch x and y*: student switched the x-axis with the y-axis. *3) Move only in one dimension* student moved the correct distance in either x or y, but remained in zero for the other dimension. The system could use this information to address misconceptions individually.

Multiple Paths to a Solution. The behavior graph enables a visualization of the various paths taken by students to get to the answer (both correct and incorrect), with thicker edges indicating more students took a given path. Therefore, we could identify that most students preferred to move positive distances instead of negative distances and generally turned using cardinal points instead of angles. The system could use this information to prompt students to consider alternative paths.

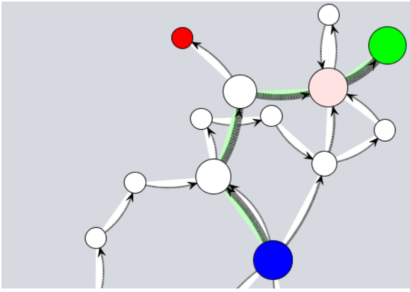

Fig. 3. Behavior graph with video information. A green edge indicates more embodiment

Influence of Embodiment. We also used the tool to encode video information into the log files. For this analysis, we encoded information from a single problem that produced a variety of correct and incorrect solution strategies from participants who interacted with the system. Within this problem, we annotated log data with participants' levels of embodied movement. Subject movement was coded using a binary schema: At each opportunity for physical interaction, a score of either 0 (little/no movement) or 1 (purposeful movement) was added as an annotation to the subject's log data. These annotations added a green highlighting to the edges of the behavior graph that denoted a higher level of movement. As illustrated in Fig. 3, by looking at the subject's behavior at each step in the problem solving process, we can identify a higher average level of embodied movement and behaviors occurring on transitions that are part of correct solution paths. This exploratory visualization may indicate an interesting relationship between levels of embodiment and problem-solving success. We see this analysis as a jumping-off point for quantitative analysis of the relationship between embodiment in our system and problem-solving success.

4 Conclusion and Discussion

In this paper, we presented a tool that facilitates analysis by integrating data from logs and video into a behavior graph. The features of this tool were demonstrated using data from a study that used the TAG System. Using the graph generated by the tool, we identified strategies, misconceptions and multiple solution paths. Furthermore, the encoded video information provided visual insight on aspects such as the relationship between a student's movements and their efficiency in solving the posed problems.

Future research could focus on making this tool generalizable, enabling it to work on systems that use different log structures. Different forms of data visualization could also improve its usefulness, such as making use of the temporal aspect of the data, allowing users to see the evolution of the graphs. Lastly, clustering algorithms would be a natural step towards automating the analysis of this data, thus improving the speed through which conclusions could be drawn from the graph.

Acknowledgments. This research was funded by NSF 1249406: EAGER: A Teachable Robot for Mathematics Learning in Middle School Classrooms and by the CAPES Foundation, Ministry of Education of Brazil, Brasília - DF 70040-020, Brazil.

References

1. Romero, C., Ventura, S., García, E.: Data mining in course management systems: Moodle case study and tutorial. Comput. Educ. 51, 368–384 (2008)
2. Romero, C., Ventura, S.: Educational data mining: A survey from 1995 to 2005. Expert Syst. Appl. 33, 135–146 (2007)
3. Arroyo, I., Woolf, B.: Inferring learning and attitudes from a Bayesian Network of log file data. In: AIED, pp. 33–40 (2005)
4. Lau, T., Horvitz, E.: Patterns of Search: Analyzing and Modeling Web Query Refinement. Courses Lect. Cent. Mech. Sci., 119–128 (1999)
5. Mostow, J., Beck, J.: Some useful tactics to modify, map and mine data from intelligent tutors. Nat. Lang. Eng. 12, 195–208 (2006)
6. Murray, T.: Authoring knowledge-based tutors: Tools for content, instructional strategy, student model, and interface design. J. Learn. Sci. 7, 5–64 (1998)
7. Harrison, B., Baecker, R.: Designing video annotation and analysis systems. Graph. Interface. 92, 157–166 (1992)
8. Costa, M., Correia, N., Guimarães, N.: Annotations as multiple perspectives of video content. In: Proc. Tenth ACM Int. Conf. Multimed., pp. 283–286 (2002)
9. Kipp, M.: ANVIL: A Generic Annotation Tool for Multimodal Dialogue. In: Proceedings of the 7th European Conference on Speech Communication and Technology (Eurospeech), Aalborg, Denmark, pp. 1367–1370 (2001)
10. Newell, A.: On the analysis of human problem solving protocols (1966)
11. McLaren, B., Koedinger, K., Schneider, M.: Bootstrapping Novice Data: Semi-automated tutor authoring using student log files (2004)
12. Muldner, K., Lozano, C., Girotto, V., Burleson, W., Walker, E.: Designing a Tangible Learning Environment with a Teachable Agent. In: Lane, H.C., Yacef, K., Mostow, J., Pavlik, P. (eds.) AIED 2013. LNCS, vol. 7926, pp. 299–308. Springer, Heidelberg (2013)
13. Muldner, K., Girotto, V., Lozano, C., Burleson, W., Walker, E.: The Impact of a Social Robot's Attributions for Success and Failure in a Teachable Agent Framework Tangible Activities For Geometry (TAG). In: International Conference of the Learning Sciences (2014)