Using Epistemic Networks with Automated Codes to Understand Why Players Quit Levels in a Learning Game

Shamya Karumbaiah[1], Ryan S Baker[1], Amanda Barany[2], and Valerie Shute[3]

[1]University of Pennsylvania, Philadelphia PA 19104, USA
[2]Drexel University, Philadelphia PA 19104, USA
[3]Florida State University, Tallahassee FL 32306 USA
shamya@upenn.edu

Abstract. Understanding why students quit a level in a learning game could inform the design of appropriate and timely interventions to keep students motivated to persevere. In this paper, we study student quitting behavior in Physics Playground (PP) – a Physics game for secondary school students. We focus on student cognition that can be inferred from their interaction with the game. PP logs meaningful and crucial student behaviors relevant to physics learning in real time. The automatically generated events in the interaction log are used as codes for quantitative ethnography analysis. We study epistemic networks from five levels to study how the temporal interconnections between the events are different for students who quit the game and those who did not. Our analysis revealed that students who quit over-rely on nudge actions and tend to settle on a solution more quickly than students who successfully complete a level, often failing to identify the correct agent and supporting objects to solve the level.

Keywords: Learning Game, Quitting Behavior, Epistemic Network Analysis, Automated Codes, Interaction Log.

1 Introduction

Digital games are increasingly used as a learning platform and are designed to keep students engaged in a fun experience while learning [1]. Such serious games need to balance the difficulty level to promote both learning and engagement – goals which may be contradictory at times [2]. In increasing the difficulty level to improve learning, there is a risk of frustrating students or even causing them to give up [2]. A student may quit a game level for many reasons. For instance, a student may find themselves unable to make progress due to a lack of conceptual understanding, difficult game mechanics or interface design (see [3] for the impact of conceptual understanding and game mechanics on student frustration). Alternatively, a player may quit a level in order to search for an easier level, a behavior also seen in intelligent tutoring systems that give the learner choice (e.g. [4]). Understanding why students quit a game informs the design of relevant and timely interventions that could keep the student motivated and prevent frustration from leading the student to give up. Thus, in this paper, we study why students quit a learning game, with a focus on the student cognition that can be inferred from their interaction with the game.

We study this question using the methods of Quantitative Ethnography [5]. Quantitative Ethnography (QE) attempts to better understand learning and learner choice using big data for thick description – a qualitative explanation of the how and why of human experiences. QE relies on creating meaningful codes – elements that are important to understand the phenomena being studied [5]. Finding ways to create good quality automated coding techniques becomes essential when working with huge volumes of learner interaction data. One solution is to explore events that are automatically generated in the interaction log data. Event-driven logging is a popular technique in software systems. In a well-designed logging system, user interaction events are defined based on the user behaviors that are most meaningful and crucial in the context of the system [6]. In this paper, we use the events logged by the learning game as automated codes relevant to student cognition, to study how the temporal interconnections between the events are different among the students who quit the game and those who did not to understand why students quit.

2 Context

Physics Playground (PP) is a learning game designed for secondary school students to understand qualitative physics [1]. In Physics Playground, students are given a sequence of two-dimensional puzzles (levels) where they need to guide a green ball to hit a red balloon (See Fig. 4). Players analyze the objects they see on the screen and draw the solution on the screen in the form of objects - often simple machines or agents (See Fig. 2). Laws of physics relating to gravity and Newton's laws apply to all objects given and drawn on the screen. Successfully completing levels includes understanding concepts such as Newton's law of force and motion, potential and kinetic energy, and conservation of momentum. There are 74 sketching levels in the game (see Fig. 1 for examples) and each level is designed to be optimally solved by particular agents. PP is nonlinear; students have complete choice in selecting levels.

Participants in this study consisted of 137 students (57 male, 80 female) in the 8th and 9th grades enrolled in a public school with a diverse population in a medium-sized city in the southeastern U.S. The game content was aligned with state standards relating to Newtonian Physics. The study was conducted in a computer-enabled classroom with 30 desktop computers over four consecutive days. The software logged all the student interactions in a log file.

A quit prediction model was previously built for this game with the goal to identify potential moments where cognitive support could support a struggling student in developing their emerging understanding of key concepts and principles [7]. While predicting when a student is likely to quit is important, it is also crucial to understand why the student is likely to quit in order to inform the design of supports that address students' individual needs. Studying why students quit could also give insights on how to improve the design of the features used in the model, potentially improving model performance. Since our eventual goal is to use our insights to improve a prediction model (and validate that model on new data), we restrict this paper's analysis to an arbitrarily chosen 20% of our students.

3 Method

The unit of analysis in this study is a level in the learning game. We have selected five levels with a high percentage of quitting. These five levels vary across Physics concept involved in the level, the agent expected to be used to solve the level (See Fig. 2), and difficulty (see Fig. 1, Table 1, 2).

Table 1. Description of the five levels chosen for analysis. The Physics concepts involved are N1stL (Newton's First Law), EcT (Energy can Transfer) and PoT (Properties of Torque).

Level	Playground (PG)	Agent	Primary Concept	Secondary Concept	Solution Video
Flower Power	PG #4	Ramp	N1stL	EcT	tiny.cc/flwr
Big Watermill	PG #4	Ramp	N1stL	EcT	-
Caterpillar	PG #4	Springboard	EcT	-	-
Need Fulcrum	PG #3	Lever	PoT	EcT	tiny.cc/flcrm
Shark	PG #3	Lever	PoT	EcT	tiny.cc/shrk

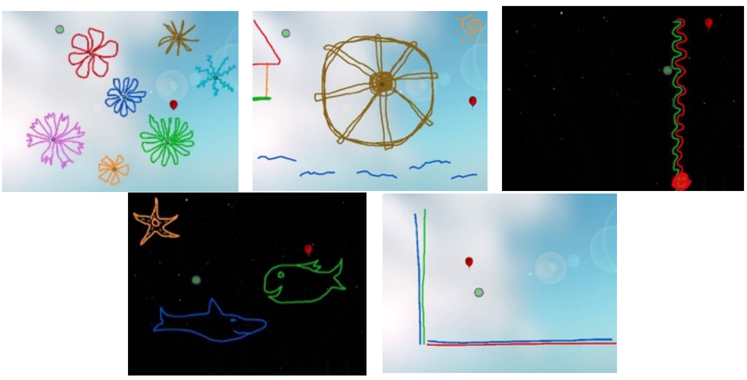

Fig. 1. The five levels of Physics Playground analyzed in this paper. Top row (left to right) – Flower Power, Big Watermill, Caterpillar. Bottom row (left to right) - Shark, Need Fulcrum

Table 2. Quit levels and difficulty levels of the five levels chosen for analysis.

Level	Game Mechanics	Physics Understanding	#Students	%Quit
Flower Power	4	2	20	35.45
Big Watermill	3	2	23	43.70

Caterpillar	4	2	22	40.24
Need Fulcrum	1	4	25	42.41
Shark	4	4	22	44.14

3.1 Epistemic Network Analysis using Automated Codes of Student Gameplay

We study learning within this serious game by interpreting students' activity as expressed in the record of their interaction with the game. Event-based logging automatically captures how and when students use the human-created codes within the tool. For instance, in PP, students draw symbolic representations of various objects like agents (e.g. lever, ramp), fulcrums, mass, and pins. An agent identification system was developed in the game to infer the type of agent drawn based on features like the presence of an arm, number of pins attached, and direction of its movement (see [1] for more details). This system has a 95% accuracy when compared with human ratings. The logging mechanism uses this system to automatically detect the creation and modification of agents in real-time from the student's sketch and logs these as events with timestamps. Studying the temporal interconnection between events using Epistemic Network Analysis (ENA; [8]) gives us a way to better understand students' cognition. We examine how the events are related differently to one another in the group of students who quit a game level as compared to the ones who did not. In this analysis, we are focusing on the following five automatically coded events, detected as indicated below:

1. *Agent Creation* (see Fig. 2)
   a. *Ramp* - detection of an object that a ball rolls along, across the screen
   b. *Springboard* – detection of an object that is attached to two or more pins and rotates to propel the ball upward
   c. *Pendulum* – detection of an object that rotates on a single pin
   d. *Lever* – detection of a secondary object that falls on a primary object, which in turn rotates on a fulcrum (a support on which a lever pivots) to launch the ball.
2. *Draw.Freeform* – the creation of a freeform object (including the agents)
3. *Draw.Pin* – the creation of a pin object
4. *Erase* – the erasure of a freeform or pin object
5. *Nudge* - the player clicked on the ball to move it to the left or right

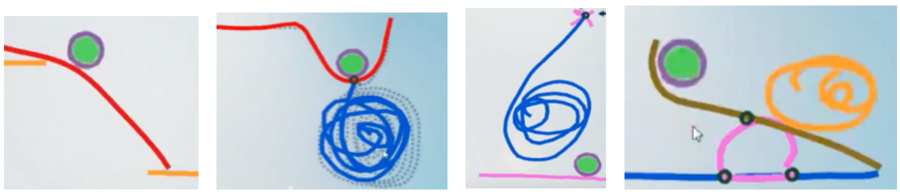

Fig. 2. The four agents used to solve PP levels. *Left to right* – ramp, springboard, pendulum, lever.

An epistemic network for a game level is created from the temporally sequenced one-hot encodings of the events that occurred during the students' gameplay. We segment the data by the student since a single students' attempt in a level is an appropriate unit of interconnected behaviors. On an average, PP logs an event once every 2 seconds. We chose a moving window size of 10 which on average corresponds to 20 seconds of gameplay. Due to the fine-grained nature of the event logs, we chose a relatively high moving window size than in many ENA analyses (e.g. [9, 10]) to get an appropriate temporal context to identify relevant co-occurrences of events (e.g. mass drawn and erased to find the needed weight, pins drawn and erased to place a springboard at a precise position, mass dropped on a lever to launch the ball).

4 Results

Figure 3 presents the difference networks for the five levels examined. These networks highlight the most salient difference between the epistemic networks of students who quit the level and those who did not. Along the X-axis (dimension 1 after means rotation), a two-sample t-test assuming unequal variance showed the group who quit was statistically significantly different from the group who did not quit at alpha=0.05 with effect size of d=1.38 for *Flower Power*, d=1.31 for *Big Watermill*, d=0.84 for *Caterpillar,* d=1.29 for *Shark* and d=0.84 for *Need Fulcrum.* The difference is not statistically significant along the Y-axis (p=1, d=0). Next, we present four key themes that provide us insights on why students quit a level in PP.

4.1 Missing Agent Identification

One of the key steps in solving a level in PP is to identify the agent needed. Across all the five networks in Figure 3, the students who quit the level without solving it did not use the agent involved - a ramp for *Flower Power* and *Big Watermill*, a springboard for *Caterpillar*, and a lever for *Shark* and *Need Fulcrum* – and other events. By contrast, students who solved the level successfully used the agent involved - except in *Need Fulcrum* (this is explained further in the next theme). Since the event *Draw.Freeform* encompasses the drawing of any freeform object including all agents, it is expected to be the event with the most connections. Apart from *Need Fulcrum*, the strongest connection out of *Draw.Freeform* for the students who did not quit a level is to the agent expected to be needed in that level. Quitting in this case can be attributed to the lack of conceptual understanding of physics.

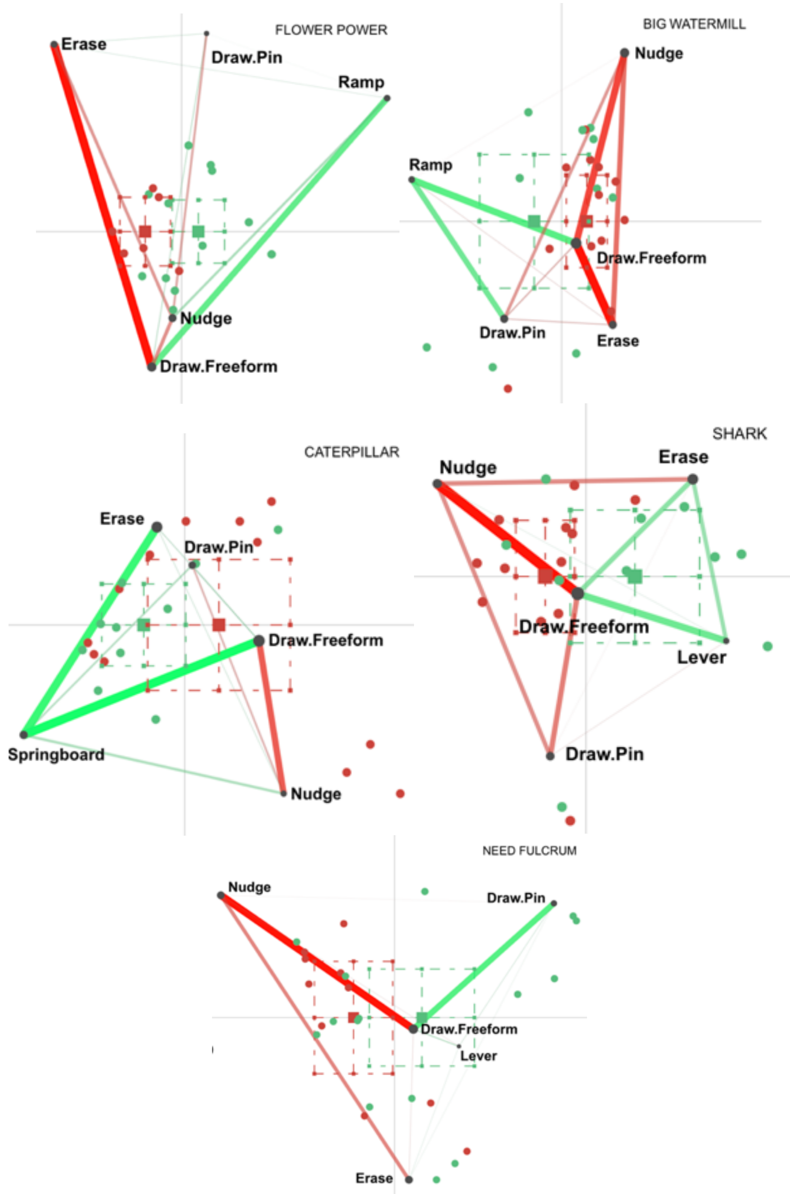

Fig. 3. The difference networks corresponding to the five levels examined. From top to bottom and left to right – *Flower Power, Big Watermill, Caterpillar, Shark, Need Fulcrum.* In these networks, red connections are made more frequently by students who eventually quit the level without solving it, while green connections are made more frequently by those who complete the level.

4.2 Missing Identification of Supporting Objects

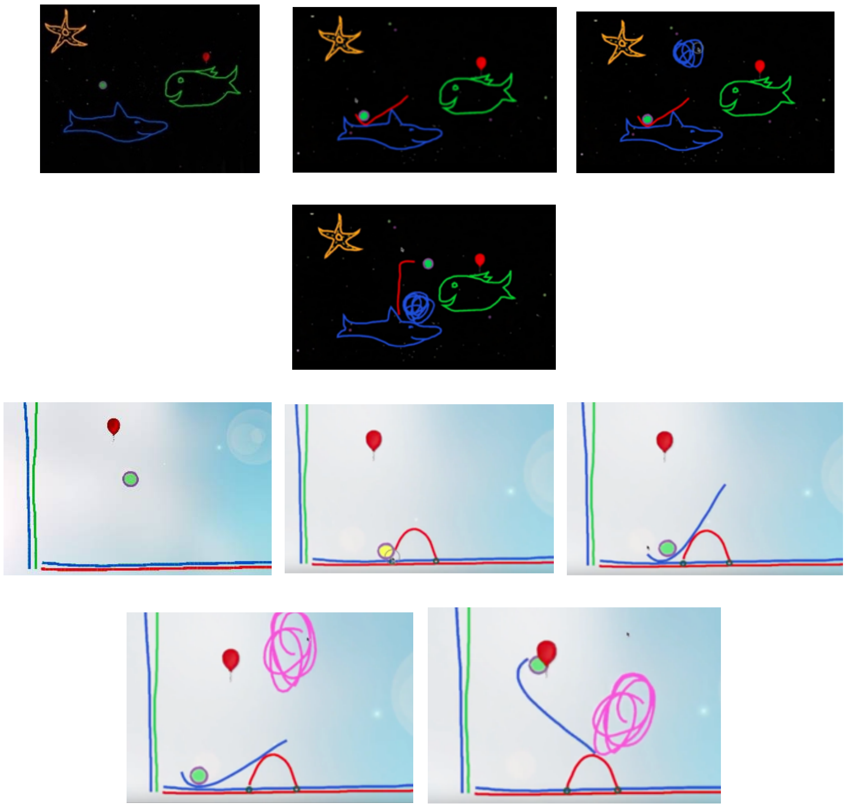

Fig. 4. Solution for *Shark* (top) as compared to *Need Fulcrum* (bottom). Note the extra action (second step) of adding a fulcrum in *Need Fulcrum*.

Comparing connections in the pairs of networks involving the same agent helps us highlight another missing cognitive connection among the students who quit. The first pair is *Flower Power* and *Big Watermill* – both of which use a ramp in their solutions. However, whereas the ramp can rest on one of the flowers in *Flower Power* (See Fig. 1), *Big Watermill* needs pins on the watermill to hold the ramp from falling off the screen. This cognitive connection can be observed in the students who did not quit *Big Watermill*. Along with a stronger connection between *Draw.Freeform* and *Ramp*, these students also have a strong connection between *Ramp* and *Draw.Pin*. Failure to hold the ramp on the screen using pins results in the ramp being undetected even for the students who may have identified the right agent but eventually quit the level due to missing pins. Quitting in such cases can be attributed to the difficult game mechanics instead of the student lacking conceptual understanding in Physics.

The second pair is *Shark* and *Need Fulcrum* – both of which needs a lever to solve. In *Shark* (See Fig. 4), a lever resting on the blue Shark's fin can catch the falling ball and launch it to the balloon when a mass is dropped. By contrast, *Need Fulcrum* (as the name suggests) needs an additional object – a fulcrum (the red arch in Fig. 4 bottom row). While the identification of the agent needed may not be difficult, the idea of using a fulcrum by fixing it on the plane using pins is the missing cognitive connection among the students who quit the level without solving it. This can be seen in the difference network as the lack of a strong connection between *Draw.Freeform* and *Draw.Pin* among the students who quit the level without solving it as compared to the students who successfully complete the level.

4.3 Over-reliance on Nudge

Four out of five networks (See Fig. 3) show a stronger association between *Draw.Freeform* and *Nudge* among the students who did not solve the game level. Nudging the ball with little connection to agent or pin creation events indicates repeated attempts at controlling the ball without creating meaningful objects that correspond to the physics concepts involved. In some cases, this could indicate wheel spinning [11] where students play for substantial amounts of time without making progress and eventually quit the level unsolved. The only other event that is closely associated with *Nudge* is *Erase*. This includes cases where the students are trying solutions that are completely incorrect or where they may have identified the agent conceptually but are unable to implement the solution in the game due to the missing identification of supporting objects or complex game mechanics. It is also interesting to observe the little to no connection to *Nudge* from any of the events in the students who solve the five levels successfully.

4.4 Limited Early Action Expansion and Later Action Convergence

There are also differences in the early and late behaviors between students who quit a level and those who didn't. The average and standard deviations (in parenthesis) of time spent per level (in minutes) for the group that quit and that which didn't are comparable - 3.73 (0.59) vs 4.08 (0.65) for *Flower Power,* 3.97 (0.75) vs 4.07 (0.76) for *Big Watermill,* 3.95(0.66) vs 3.90 (0.81) for *Caterpillar,* 4.25 (1.07) vs 4.65 (0.97) for *Shark,* 4.20 (1.15) vs 4.21 (1.14) *Need Fulcrum.* However, when students' attempts were divided into different time quartiles, we see that students who eventually quit a level do limited action exploration in the beginning (often not involving agents) and continue to produce the same limited event set for the rest of the attempt. By contrast, the students who eventually solve the level start with an expanded search for possible actions and continue to converge on a smaller subset of actions as their gameplay goes on. Figure 5 illustrates this by comparing two students – one who solved *Need Fulcrum* successfully and one who did not. As we see in the three red networks, the student who quit started by frequently connecting three events – *Draw.Freefrom, Erase* and *Nudge* and continued to follow the same approach until they quit. In contrary, the student who solved the level began with more distributed exploration and converged to just *Draw.Freeform* and *Erase* – in this case indicating the final attempts to find the correct weight to drop on the lever to launch the ball.

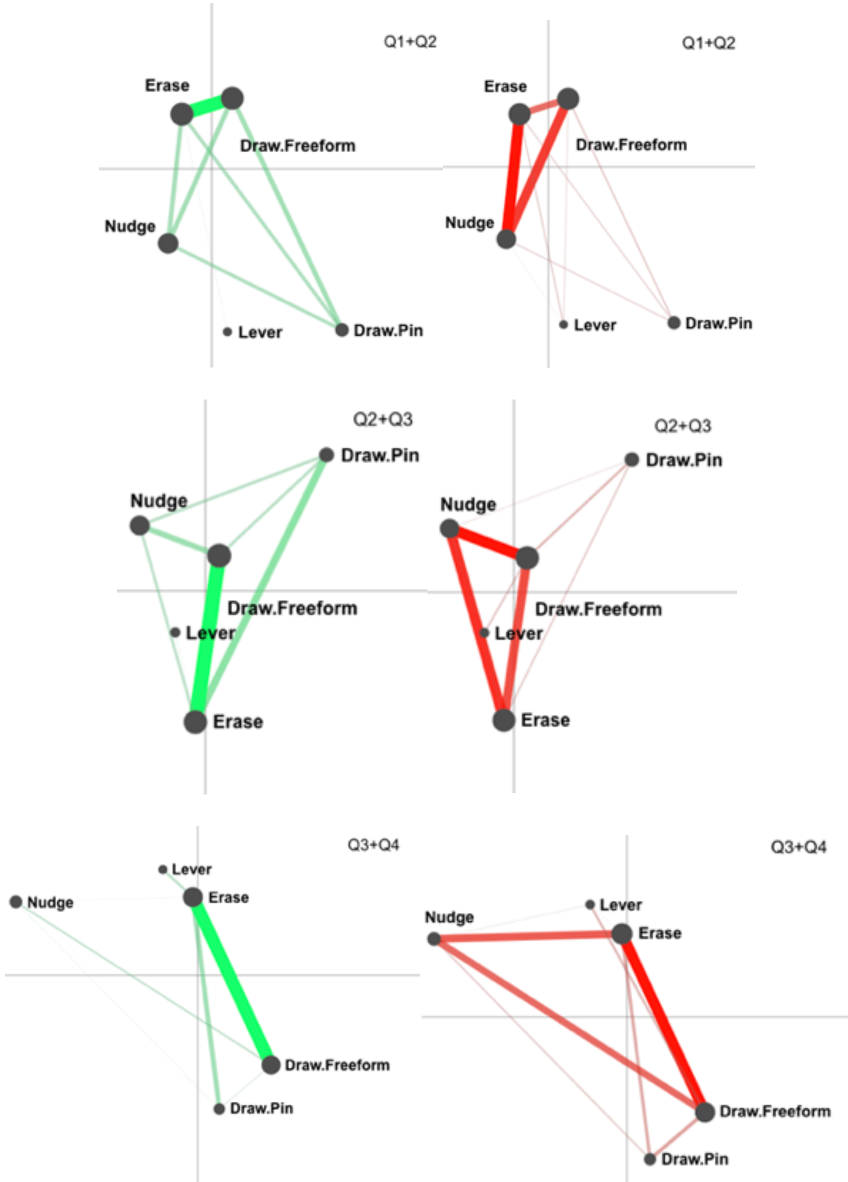

Fig. 5. Comparing early (left) to late (right) epistemic networks of a student who successfully solved *Need Fulcrum* (in green) to a student who quit (in red)

5 Discussion

In summary, we used Epistemic Network Analysis (ENA) to generate thick description of students' quitting behavior in a Physics learning game called Physics Playground

(PP). We do so using the automatically generated events from the interaction log as codes for the quantitative ethnography analysis. Across the five levels investigated, our analysis revealed a set of themes which point at some potential root causes for why students quit levels unsolved in a learning game. In some cases, students may not identify the agent needed to correctly solve the level, indicating their lack of conceptual understanding of Physics. In other cases where students appear to have identified an appropriate agent, they may struggle with the difficult game mechanics around placing supporting objects or timing and precision in the placement of the objects. Other students may display wheel spinning behavior where they just nudging the ball repeatedly without making any progress. These insights are valuable to design appropriate interventions for individual students' needs. For instance, cognitive supports could be provided for the needed conceptual understanding or supports could be incorporated into the play interface for difficult game mechanics.

One limitation in this analysis is that the students in this dataset are of similar ages and live in the same region. Hence, it will be important to test the generalizability of our findings on data from a broader and more diverse range of students. In our future work, we also plan to use ENA to improve the quit prediction model which was built to deliver the interventions in a timely manner. This analysis can inform the engineering of new features that capture the behavior differences like use of a relevant agent, use of nudge and action convergence over time. We also may be able to use the epistemic networks directly for quit prediction. When a new student works on a level, we could look for whether their gameplay converges to the previously observed networks of students who solved the level successfully or those who did not.

In this paper, we have demonstrated an approach of using ENA with automated codes that has the potential to be applied in other intelligent tutoring systems with well-designed event-based logging mechanism to study constructs related to student learning, engagement, and experience in the system. It is worth noting that while this paper has primarily focused on student cognition inferred through their interaction, this does not eliminate the possibility that there could be other broader social and cultural reasons influencing factors such as students' interest, motivation, and their perceptions or beliefs about competence, that might in turn lead to students' quitting behavior. No model is perfect, but what we have learned from ENA on interaction data has the potential to focus our efforts to enhance how we support students within the scope of the game.

References

1. Shute, V. J., Ventura, M., Kim, Y. J.: Assessment and learning of qualitative physics in newton's playground. The Journal of Educational Research, 106(6), 423-430 (2013).
2. Lomas, D., Patel, K., Forlizzi, J. L., Koedinger, K. R.: Optimizing challenge in an educational game using large-scale design experiments. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 89-98, ACM, New York, NY (2013).
3. Karumbaiah, S., Rahimi, S., Baker, R.S, Shute, V. J., D'Mello, S. Is student frustration in learning games more associated with game mechanics or conceptual understanding? In: Kay, J., Luckin, R. (eds.) 13th International Conference of Learning Sciences, vol. 3, pp. 1385-1386, London, UK (2018).

4. Baker, R. S., Mitrović, A., Mathews, M.: Detecting gaming the system in constraint-based tutors. In: De Bra, P., Kobsa, A., Chin, D. (eds.) 18th International Conference on User Modeling, Adaptation, and Personalization, pp. 267-278. Springer, Berlin, Heidelberg (2010).
5. Shaffer, D. W.: Quantitative ethnography. Cathcart Press (2017).
6. Owen, V. E.: Capturing in-game learner trajectories with ADAGE (assessment data aggregator for game environments): A cross-method analysis. Doctoral dissertation, University of Wisconsin-Madison, Madison, WI (2014).
7. Karumbaiah, S., Baker, R. S., Shute, V.: Predicting quitting in students playing a learning game. In: Boyer, K. E., Yudelson, M. (eds.) Proceedings of the 11th International Conference on Educational Data Mining, pp. 167-176 (2018).
8. Shaffer, D. W., Collier, W., Ruis, A. R: A tutorial on epistemic network analysis: Analyzing the structure of connections in cognitive, social, and interaction data. Journal of Learning Analytics, 3(3), 9-45 (2016).
9. Arastoopour, G., Shaffer, D. W., Swiecki, Z., Ruis, A. R., Chesler, N. C.: Teaching and assessing engineering design thinking with virtual internships and epistemic network analysis. International Journal of Engineering Education, 32(2) (2016).
10. Knight, S., Arastoopour, G., Williamson Shaffer, D., Buckingham Shum, S., Littleton, K.: Epistemic networks for epistemic commitments. In: Polman, J. L., Kyza, E. A., O'Neill, K., Tabak, I., Penuel, W. R. Jurow, A. S., O'Connor, K., Lee, T., D'Amico, L. (eds.) Learning and Becoming in Practice: The International Conference of the Learning Sciences, vol. 1, pp. 150-157, Boulder, CO (2014).
11. Beck, J. E., Gong, Y.: Wheel-spinning: Students who fail to master a skill. In: Lane, H. C., Yacef, K., Mostow, J., Pavlik, P. (eds.) Proceedings of the 16th International Conference on Artificial Intelligence in Education, pp. 431-440, Memphis, TN (2013).