Modeling and Analyzing Inquiry Strategies in Open-Ended Learning Environments

Tanja Käser & Daniel L. Schwartz

International Journal of Artificial Intelligence in Education
Official Journal of the International AIED Society

ISSN 1560-4292

Int J Artif Intell Educ
DOI 10.1007/s40593-020-00199-y

Springer

International Journal of Artificial Intelligence in Education
https://doi.org/10.1007/s40593-020-00199-y

ARTICLE

Modeling and Analyzing Inquiry Strategies in Open-Ended Learning Environments

Tanja Käser[1] · **Daniel L. Schwartz**[2]

Published online: 09 September 2020

Abstract

Modeling and predicting student learning in computer-based environments often relies solely on sequences of accuracy data. Previous research suggests that it does not only matter what we learn, but also how we learn. The detection and analysis of learning behavior becomes especially important, when dealing with open-ended exploration environments, which do not dictate prescribed learning sequences and skills. In this paper, we work with data collected from an inquiry-based environment. We demonstrate that 1) students' inquiry strategies indeed influence the learning outcome, and 2) students' inquiry strategies also seem to be predictive for their academic achievement. Furthermore, we identified a new positive inquiry strategy, which has not yet been described in the literature. We propose the use of a probabilistic model jointly representing student knowledge and strategies and show that the inclusion of learning behavior into the model significantly improves prediction of external posttest results compared to only using accuracy data, a result that we validated on a second data set. Furthermore, we cluster the children into different groups with similar learning strategies to get a complete picture of students' inquiry behavior. The obtained clusters can be semantically interpreted and are not only correlated to learning success in the game, but also to students' science grades and standardized math assessments. We also validated the cluster solution on a second data set. The inquiry-based environment together with the clustering solution has the potential to serve as an assessment tool for teachers and tutors.

Keywords Learning · Strategies · Prediction · Simulation · Probabilistic models · Clustering

✉ Tanja Käser
tanja.kaeser@epfl.ch

Daniel L. Schwartz
daniel.schwartz@stanford.edu

[1] EPFL, Lausanne, Switzerland

[2] Graduate School of Education, Stanford University, Stanford, CA, USA

Introduction

Over the last decade, there has been an increase in the use of open-ended learning environments such as discovery environments (Shute and Glaser 1990), narrative-centered learning environments (Rowe et al. 2009), or simulations (Wieman et al. 2008). Ideally, students explore different configurations of parameters to infer the underlying principles. One rationale is that students learn the principles more deeply through exploration than if they are simply told the principles and asked to practice applying them (Schwartz et al. 2011). However, not all students apply the inquiry skills necessary to effectively explore the environment (Kinnebrew et al. 2013; Sabourin et al. 2013; Mayer 2004). Modeling students' learning as they try to benefit from relatively open-ended inquiry environments may be a useful next step in our abilities to support students' development of independent capacities to learn from simulations and other discovery environments.

Intuitively, teachers and intelligent systems should be able to observe students' learning strategies to make formative suggestions to improve learning. In practice, this can be very difficult, especially in the context of students trying to induce new rules or principles. Success at observing student learning strategies requires closely observing how a student goes about learning, and it requires tasks that support student opportunities to apply strategies conducive to rule induction. It is much easier to observe whether a student is achieving a correct or incorrect answer than to observe the learning strategies. To help clarify this point, we consider two examples as thought experiments.

First, imagine a math tutor working with a student. The math tutor sees that the student has made a mistake. A good tutor will track down what caused the mistake to identify what knowledge the student is missing or misunderstanding. Then, typically, the tutor will explain the right actions and why, and then provide the student with a new practice problem targeting the new instructional content. Notice that in this first example, the tutor is observing closely, but the tutor never observes the student's learning strategies. Instead, the tutor observes problem-solving steps and whether the student offers right or wrong answers. Moreover, the tutor takes charge of the student learning, so that any student learning strategies do not readily reveal themselves.

Second, consider a scenario where the learning context is more conducive to observing student learning strategies. In this case, the student is working with an open-ended learning environment and trying to induce patterns and laws of nature from an interactive science simulation. The student sets different parameters within the simulation to see how it changes outcomes. Ideally, the student finds the correlations among variables, and perhaps even infers causal explanations. In this context, the tutor cannot rely solely on right and wrong answers, because the student is not responding to specific questions. Instead, the tutor may try to figure out what strategies the student is using to be helpful. This depends on the tutor having a strong catalog of possible learning strategies for open-ended environments; a knowledge of which ones are actually effective strategies; and, the ability to recognize which behaviors with the interactive simulation correspond to which strategies. For instance, on the one hand the tutor might notice that the student is exploring boundary conditions

by setting extreme values, but on the other hand, the tutor might notice that the student is not employing a control of variable strategy by systematically varying a single parameter at a time. Given this observation, the tutor could help the student learn to use control of variables and improve the learning strategies available to the student, which in turn, would help the student learn about the phenomena portrayed in the simulation.

Our goal is to bring artificial intelligence to the second example. Interactive, and relatively open-ended, simulations are beginning to suffuse education (e.g., as of March 2019, the PhET simulations project (Wieman et al. 2008) alone, has delivered 650 million simulations). It is intractable for a teacher to observe a classroom of children doing heads-down interactions with simulations and infer their learning strategies. Moreover, the validated catalog of effective strategies remains impoverished, in part, because learning from simulations is relatively new, compared to learning from textbook problems. Artificial intelligence may provide a way to augment teachers' abilities to identify and take action on students' use of learning strategies in these complex, free-flowing environments. In this paper, we focus on the first half of this proposition using artificial intelligence to help identify learning. For example, can the computer help detect learning strategies? Can we determine which learning strategies are indeed good for learning? Can we characterize them transparently so that a teacher could conceivably use them as formative assessments?

Specifically, we combine accuracy and strategy data to develop student models that not only predict learning outcomes within the environment, but also predict student performance outside of the environment. As we demonstrate, the analysis of how students go about learning is particularly useful for predicting performance outside the environment, when the presentation of problems looks different. We also show that clustering students based on strategy and accuracy can reveal important information about how students choose to learn, which appears to be significant for how they do in school more generally. To do the work, we rely on an inquiry environment designed to help capture information about student accuracy when solving pre-defined problems, while also tracking their strategies for learning the relevant information in this environment. The environment is an adaptation of an assessment framework that Schwartz and Arena (2013) called 'choice-based assessments'. The assessments leave it up to the student to choose whether and how to learn. The assessment goal is to determine which strategic choices students spontaneously make when they are not receiving strong guidance and to determine whether specific choices are better for learning than other choices. The choice-based assessments, which are typically presented as games to the students, are each designed to assess specific classes of learning strategies, including critical thinking (Chi et al. 2014), consultation of literature (Chin et al. 2016), and feedback seeking behavior (Cutumisu et al. 2015). In the current instance, we used a choice-based assessment called `TugLet`, which is a 10-15 minute interactive game. `TugLet` differs from prior choice-based assessments, because it also includes specific questions that students must answer correctly to win the game. It is a hybrid environment – in one 'room', students can engage in inquiry by using a simulation, and in another 'room', they can try to solve specific problems and receive right-wrong feedback.

In our first contribution, we propose that looking at how students go about their learning may provide useful information that is not fully captured by their right/wrong answers. For instance, some students may try to finish as quickly as possible, whereas others may take steps to understand. While both would eventually attain high accuracy within an environment, those who take steps towards understanding may learn the principles that determine when an answer is right or wrong. Our extensive analysis of the collected log data demonstrates that students' exploration choices and strategies significantly influence the learning outcome. We have also identified a new positive inquiry strategy, which has not been previously described in the literature. The identified strategy is what separates the top performing students from the others and is correlated to academic achievement. To bridge knowledge, operationalized as accuracy, and learning choices, we present a novel technique for creating simple probabilistic student models jointly representing student knowledge and strategies. We evaluate the models' prediction accuracy within the computer-based game as well as on an external posttest. Our results demonstrate that modeling student knowledge and student strategies at the same time significantly improves predictive performance and therefore constitutes an improved representation of learning compared to representing knowledge alone. Moreover, we replicate these results on a new validation data set.

In our second contribution, we cluster students into groups with different inquiry strategies. The obtained clusters are semantically interpretable. For example, one cluster captures students who use rapid trial-and-error inquiry, whereas another cluster captures students who simply want to beat the game. The clusters not only predict how well students do on a subsequent, out-of-game posttest, they also predict students' academic achievements in terms of science grades and scores in standardized math assessments. Moreover, the cluster solution is stable, as the validation on a second independent data set demonstrates. `TugLet` along with the presented clustering algorithm therefore has the potential to serve as an assessment tool for teachers and tutors, ideally providing them new insights into students' inquiry skills, and ideally, helping them to improve students' learning strategies.

Related Work

Up to now, research in detecting and analyzing student learning has focused on accurately representing and predicting student knowledge based on the students' past accuracy within the computerized learning environment, i.e., the students' answers to tasks are assessed and serve as observations for the respective method. One of the most popular approaches to representing and predicting student knowledge accurately is Bayesian Knowledge Tracing (BKT) (Corbett and Anderson 1994), a technique that has been continuously improved over the years, e.g., Pardos et al. (2012), Wang and Beck (2013), and Yudelson et al. (2013). Other techniques are based on item response theory, such as the additive factors model (Cen et al. 2007; 2008) or performance factors analysis (Pavlik et al. 2009). Furthermore, dynamic Bayesian networks have been used to represent and predict student knowledge, e.g., González-Brenes and Mostow (2012) and Käser et al. (2014).

The aforementioned approaches describe student knowledge as a set of skills or knowledge components used to solve problems, and the way to infer student knowledge is to determine whether they are correct in problem solving. The decomposition and tracking of fine-grained knowledge acquisition is a major strength of these techniques. Open-ended inquiry environments, however, can be problematic for these approaches. It is difficult to assign an accuracy parameter when students are freely experimenting with a simulation rather than responding to 'right-wrong' problems. At the same time, inquiry environments may yield significant information about students' learning strategies, as they try to figure out the rules and principles that govern the phenomena portrayed in the environment.

Previous research has demonstrated that features such as learning choices, styles, and strategies influence the learning outcome. Gaševic et al. (2017) clustered students of an online engineering course according to their learning sequences and demonstrated that students' learning approaches were correlated to academic achievement. Furthermore, the strategies that students applied in an educational game influenced their implicit science learning (Rowe et al. 2014; Eagle and Barnes 2014). Given the freedom open-ended learning environments offer to the learner, they afford the opportunity to analyze their learning strategies, which in turn, should predict their learning outcomes. Several researchers are working on ways to extend existing student modeling approaches to capture learning strategies such as help-seeking (Beck et al. 2008; Roll et al. 2011; Roll et al. 2014) and off-task behavior (Baker et al. 2004; Baker et al. 2008) into the respective models. FAST (González-Brenes et al. 2014) is a technique for integrating general features into BKT. Dynamic mixture models (Johns and Woolf 2006) and DBNs (Schultz and Arroyo 2014) have been used to trace student engagement and knowledge in parallel.

A common goal is to detect and analyze learner behavior with the intent of increasing the adaptivity of the system. Sawyer et al. (2018) used time series to represent student trajectories through a game-based learning environment and computed the distance to an expert path to get an assessment of students' problem-solving behavior. Others (Mojarad et al. 2018; Barata et al. 2016; Fang et al. 2018) used clustering approaches to identify different types of learners. Truong-Sinh et al. (2017) investigated whether typical learning behavior in a massive open online course (MOOC) can be transferred to other courses. Zhang et al. (2017) clustered students according to their problem solving behaviors and demonstrated that students over time transitioned to better performing clusters. Recent research (Geigle and Zhai 2017) has also attempted to automatically extract student activity patterns in the form of behavior state-transition graphs from large amounts of MOOC log data using a two-layer Markov model and showed that the extracted patterns can be interpreted. Kardan and Conati (2011) have formulated the clustering idea into a student modeling framework for open-ended learning environments: students are clustered online into groups with similar learning behavior and targeted interventions are designed based on the clustering solution. When a new student interacts with the environment, the student is assigned to one of the clusters (and the corresponding intervention). The framework has for example been used to predict the mathematical learning patterns of students (Käser et al. 2013). It has also successfully been applied to an environment for learning common artificial intelligence algorithms (Amershi and Conati 2009;

Kardan and Conati 2011). Recently, Fratamico et al. (2017) used the framework to build student models for an interactive simulation of electric circuits.

The classification approach has two main advantages compared to traditional student modeling approaches. It does not depend on the explicit testing of students' knowledge components and it can ideally support individualization based on classification. Thus, on the one hand, traditional student modeling approaches allow for a more fine-grained delineation of student knowledge, as they track student accuracy at each step. On the other hand, approaches that emphasize learning behaviors and strategies provide information relevant to classifying and supporting students who take different approaches to learning on their own.

In this paper, we contribute to both the traditional student modeling and the clustering-based student modeling research strands. First, we develop a probabilistic model able to jointly represent student knowledge and strategies and demonstrate that incorporating inquiry strategies into the model significantly improves prediction of external posttest results compared to only using accuracy data. Second, we cluster the children into different groups with similar inquiry behavior. The obtained clusters can be semantically interpreted and are not only correlated to learning success in the game, but also to students' science grades and standardized math assessments.

Tuglet - the Game

To examine the relation between inquiry strategies and correct answers, we developed `TugLet`, a short interactive computer-based educational game. The topic of the game is tug-of-war. Tug-of-war is the name for the game where people grab opposite sides of a rope and try to drag the other team across a center line. The educational goal of `TugLet`, which is not revealed to the students, is to discover the underlying principles governing the tug-of-war. Students can achieve this through two different modes: *Explore* and *Challenge*.

In the *Explore* mode (illustrated in Fig. 1 (top left)), players interact with a simulation: they can set up opposing tug-of-war teams and see how they fare against each other. Each team consists of a maximum of four team members denoted by the yellow dots. Available characters for the teams come from three different force categories: large (force $f = 3$), medium ($f = 2$), and small ($f = 1$). The characters can be dragged up from the bottom of the screen onto the yellow dots to the left and right side of the carriage. Once the student is happy with the two teams, (s)he hits the play button to simulate the tug-of-war and observe the outcome. Each time the student presses the play button, we record the simulated configuration, i.e., the exact weights (and positions) of the left and right teams in a so-called tug-of-war set-up.

The *Challenge* mode tests the students' knowledge about the forces: the students predict the outcome of a tug-of-war (see Fig. 1 (bottom left)). This mode consists of eight questions constrained to have increasing complexity. These eight questions are structured as follows: one very easy question, two easy questions, two medium questions, and three hard questions. Questions are picked randomly within the given category. Questions categorized as very easy have an obvious winner, i.e., for many of these questions, the teams on the left and right sides are identical. Easy questions can

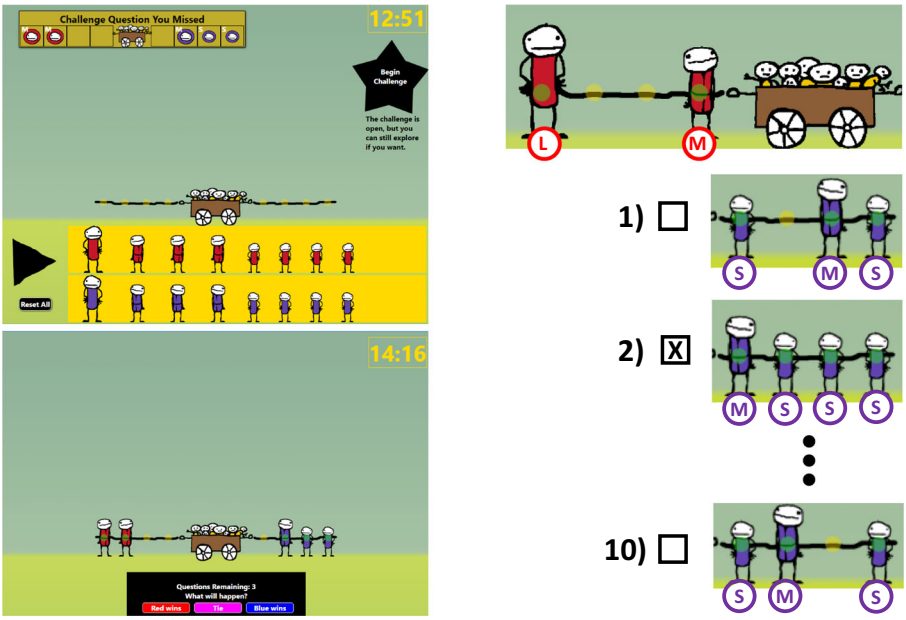

Fig. 1 The goal of the `TugLet` game is to discover the rules governing the tug-of-war by simulating different team set-ups in the *Explore* mode (top left). Students can drag up a maximum of four weights from the bottom of the screen to the left side (red) and the right side (blue) of the carriage. The tug-of-war can be simulated by pressing the `Play` button at the bottom left of the screen. Students can enter *Challenge* mode at any point in time by pressing the `Star` button at the top right of the screen. In *Challenge* mode (bottom left), students have to determine the winner of given tug-of-war configurations. The knowledge taught is assessed in the posttest (right): the left team is fixed and students have to select all the right teams leading to a tie. The second case for example results in a tie, since the force of three small (S) characters is equal to the force of one large (L) character

usually be answered intuitively, i.e., the team consisting of more characters (weights) is the winner (e.g., two medium weights on the left side versus four medium weights on the right side). Questions categorized as medium require knowledge about the relation between small and medium weights. To answer hard questions the relations between all different characters need to be understood. Once in *Challenge* mode, students get presented questions until they either make a mistake or pass the game. In case of making a mistake, the student is put back into *Explore* mode. However, students can immediately return to the *Challenge* mode without doing any explorations if they choose so.

At the beginning, all students start in a simplified *Explore* mode with only two characters (one large, one small) available per team. The *Challenge* mode is blocked for the first minute of the game. After that, students can enter the *Challenge* mode as they wish. The game is over after correctly answering the maximum of eight *Challenge* questions in a row or after a maximum play time of 15 minutes.

The free choice between *Explore* and *Challenge* mode is an important design feature of `TugLet`. The game allows us to evaluate students' choices relevant to learning. For example, do they spend all their time in the *Challenge* mode in an

attempt to beat the game, or do they choose to engage in experimentation, or some combination of the two?

TugLet comes with a short computer-based posttest measuring the knowledge acquired in the game. This allows us to determine to what extent choices and correct answers predict near transfer beyond the game itself. This is not a case of spontaneous transfer, where students need to realize which knowledge is relevant. Instead, it is a near transfer where students need to figure out how their previous learning maps to a new situation that is obviously relevant to what they have previously learned. The fact that students can 'level up' in a game does not mean they have learned well enough to use the knowledge outside of the game, even in a slightly modified format, which is one of the things we show below. The posttest assesses the students' knowledge about the forces and the relations between the different characters of the game. Students are presented with a fixed tug-of-war team for the left side and with ten different tug-of-war teams for the right side. The task is to select all the cases from the right side that will result in a tie. There are four configurations resulting in a tie. The posttest is scored as follows: each example correctly elected as a tie results in adding one point (true positive), while one point is deducted for each incorrect selection (false positive). Hence, the total posttest score corresponds to the number of true positives minus the number of false positives. The maximum possible score is therefore 4 and we set the minimum posttest score to be 0. A summary sketch of the posttest is provided in Fig. 1 (right), where 'L' denotes a large character, 'M' a medium character, and 'S' stands for a small character.

Knowledge Representation

To assess student learning and strategy use within the game, the domain knowledge needs to be formalized in a way that can accommodate different levels of student proficiency. We represent the knowledge of the students as a set of hierarchical rules describing the relations between the forces of the different characters. The complete TugLet rule set consists of $n = 12$ rules $\mathcal{R} = \{R_i\}$ with $i \in \{1, \ldots, n\}$ and is listed in Table 1. The first three rules are meta-rules defining the basic tug-of-war concepts. The remaining nine rules (rules R_4 to R_{12}) describe inequality and equality relations between the forces of the different characters. The rule set in $\mathcal{R}$ contains all the rules necessary to solve all possible configurations in the game as well as in the posttest. Note that a subset of the rules would (theoretically) be enough to derive the relations between the forces of all characters. The rules R_4, ..., R_6, R_8, and R_{10}, ...R_{12} can for example be derived from rules R_7 and R_9. The hierarchy of the rule set is necessary, because the students tend to learn in smaller steps, i.e., they test simpler hypotheses first (e.g., R_5: 'L > S'). The final rule set $\mathcal{R}$ therefore is the subset of all possible correct rules necessary to determine the winning side of all tug-of-war set-ups encountered in TugLet and in the associated posttest.

Given the set of rules, the winning side of a specific tug-of-war configuration can be determined by iteratively applying the available rules. Each tug-of-war configuration is associated with a minimum subset $\mathcal{R}_N \subseteq R$ of rules necessary to determine the winning side. The calculation of $\mathcal{R}_N$ is performed as follows: each rule $R_i \in \mathcal{R}$

Table 1 Rule set $\mathcal{R}$ representing the domain knowledge, i.e., the relationships between the different characters

Label	Rule	Description
R_1	Equality	Exact same teams on both sides result in a tie.
R_2	Cancellation	Same characters on both sides can be canceled out.
R_3	More	More characters of the same force win.
R_4	M > S	Medium character ($f = 2$) beats small character ($f = 1$).
R_5	L > S	Large character ($f = 3$) beats small character ($f = 1$).
R_6	L > M	Large character ($f = 3$) beats medium character ($f = 2$).
R_7	M = 2·S	Medium character ($f = 2$) is equal to two small characters ($f = 1 + 1$).
R_8	L > 2·S	Large character ($f = 3$) beats two small characters ($f = 1 + 1$).
R_9	L = 3·S	Large character ($f = 3$) is equal to three small characters ($f = 1 + 1 + 1$).
R_{10}	L = M + S	Large character ($f = 3$) is equal to a medium and a small character ($f = 2 + 1$).
R_{11}	2·M > L	Two medium characters ($f = 2 + 2$) beat one large character ($f = 3$).
R_{12}	S + L = 2·M	A small and a large character ($f = 3 + 1$) are equal to two medium characters ($f = 2 + 2$).

has a set of conditions attached under which this specific rule can be applied. Rule R_7 for example requires the presence of at least one medium character on the left (or right) side, respectively and a minimum of two small characters placed on the right (or left) side, respectively. To build $\mathcal{R}_N$, the system iterates through the rules $R_i \in \mathcal{R}$ and applies them, until no more rules can be applied and hence the winning side is determined. During this process, the meta rules (R_1, R_2, and R_3) as well as the simpler rules describing basic relationships between characters (e.g., R_4 or R_5) are prioritized. The resulting rule set $\mathcal{R}_N$ consists of all the applied rules. Figure 2 shows two possible solutions for the calculation of the rule set $\mathcal{R}_N$ for an ambiguous configuration, i.e., there is more than one sequence of rules leading to the correct determination of the winning side. Due to the prioritization of simpler rules, our algorithm will result in the rule set $\mathcal{R}_N = \{R_2, R_8\}$.

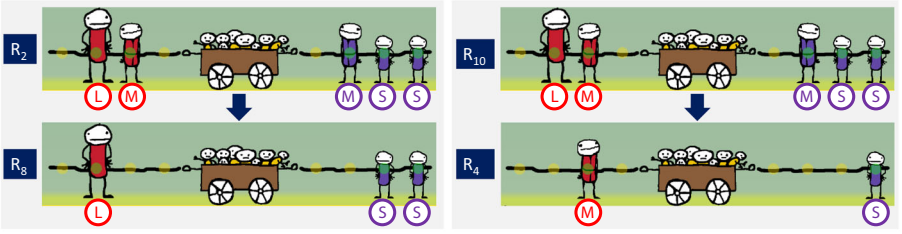

Fig. 2 Example tug-of-war configuration with one large (L), two medium (M) and two small (S) characters. The winning side can for example be determined by applying $\mathcal{R}_N = \{R_2, R_8\}$ (left) or $\mathcal{R}_N = \{L, R_4\}$ (right)

During game play, the students implicitly encounter these rules by testing out tug-of-war configurations in the *Explore* mode and by answering questions in the *Challenge* mode. We assume that each tug-of-war set-up encountered provides an opportunity for learning the principles governing the tug-of-war, i.e. the rule set. The rules, which can be acquired (or strengthened) from a given set-up are exactly the rules $R_i \in \mathcal{R}_N$ associated with the given set-up.

Experimental Setup

We collected two different data sets, an evaluation data set and a validation data set to ensure that our findings can be replicated. The set-up was the same for both studies: students played `TugLet` for a maximum time of 15 minutes, followed by a short computer-based posttest. During game play, all the user interactions were recorded in log files. The students had no prior experience with the topic from the science curriculum and had not used the PhET forces and motion simulation (Wieman et al. 2008), which was the inspiration for `TugLet` (https://phet.colorado.edu/en/simulations/category/new).

`Evaluation Data Set (ED)`. The data set used consists of 127 students (68 male, 59 female) attending the 8^{th} grade of a public middle school in California. Children come from households with medium socio-economic status (SES).

`Validation Data Set (VD)`. The data set used consists of 152 students (54 male, 83 female, 5 no information) in the 8^{th} grade of a different public middle school. The school is serving a district with families with medium to high SES.

The statistics of both data sets are listed in Table 2. Besides the total number of students per data set, we also specify variables describing the behavior and performance of students in the game. The percentage of time spent in *Explore* mode is computed as the ratio between the total amount of time spent in *Explore* mode and the total amount of time spent on the game. The number of *Challenge* questions answered

Table 2 Descriptive statistics of the `ED` and the `VD`. The maximum score in the posttest is 4. The percentage of time spent in *Explore* mode denotes the time (in s) spent in *Explore* mode divided by the total time (in s) spent in the game

Measure	`Evaluation data set`	`Validation Data Set`
Number of students	127 (54% male)	152 (36% male)
Grade	8^{th} grade	8^{th} grade
Percentage of time spent in *Explore* mode	27%	23%
Number of *Challenge* questions answered (SD)	37.2 ($\sigma = 27.3$)	35.7 ($\sigma = 26.7$)
Percentage of students passing the game	87%	97%
Average posttest score (SD)	1.9 ($\sigma = 1.6$)	2.5 ($\sigma = 1.5$)

indicates how fast students passed the game. Students from the VD performed significantly better on the posttest than students from the ED ($p < .01$). There is no significant difference in the total number of *Challenge* questions needed to pass the game between the students of the two data sets. However, students of the ED spent more time in the *Explore* mode (p = .016).

We organize the following sections according to our two main contributions.

Using the ED, we first demonstrate that a combined model of performance and strategies is better at predicting transfer than a model relying on student answers (correct/wrong) only. We do so by training different types of models on the interaction data from the the game and use them to predict students' posttest answers. This result demonstrates that passing the game does not mean that students have learned well enough to apply the knowledge outside the game. Instead, also students' learning strategies are essential. We then test our combined model on the VD and replicate the findings. The different probabilistic models and their resulting prediction accuracy on the ED have been published in 2017 (Käser et al. 2017). Here, we restate and extend these results as a basis for our analysis on the VD.

In a second step, we show that computerized inquiry environments (e.g., simulations) can reveal important information about how students choose to learn. We derive several clusters of student learning strategies and performance on the ED. The clusters can be semantically interpreted and are correlated to students' academic achievement. Furthermore, the obtained clusters and their interpretation is validated on the VD. This shows that TugLet along with the cluster solution could be used as a formative assessment by teachers.

Joint Models of Strategy and Performance

In this section, we demonstrate that including students' exploration strategies into a predictive model increases our ability to predict out-of-game performance compared to a pure knowledge model. We first present an analysis of students' exploration strategies in the game. We then build different probabilistic models based on the approach of Bayesian Knowledge Tracing (BKT) and show that a joint model of performance and strategies is best in predicting performance on the posttest.

Exploration Strategies

Most of the students (87%, $n = 111$) from the ED managed to pass the game within the given time-frame, i.e., they solved eight problems of increasing complexity in a row correctly. However, only 24% ($n = 31$) of the students had a perfect posttest. And 26% ($n = 33$) of the students had a score of 0 in the posttest. These results demonstrate that it is possible to be accurate in the game yet do unexpectedly poor out of the game.

Therefore, we investigated the trajectories of the students through the game and examined the set-ups students simulated in the *Explore* mode. We illustrate the different trajectories and exploration behaviors using three example students (Fig. 3 (top)). While student B had a perfect posttest, students A and C had a posttest score of 1 and

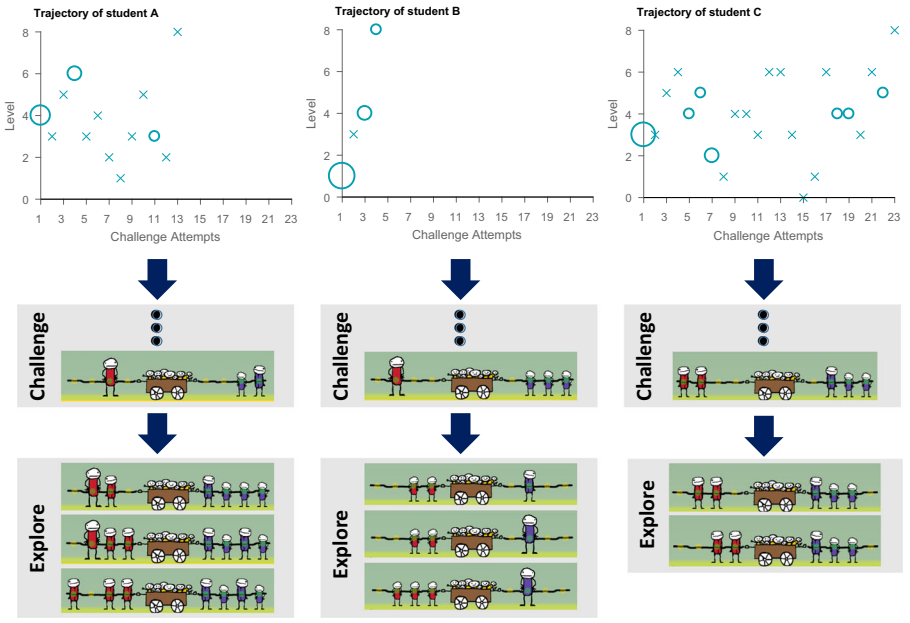

Fig. 3 Comparison of the trajectories and the simulated tug-of-war set-ups for student A (left), student B (middle) and student C (right). In the trajectory plots (top), each circle (or cross) denotes one attempt to pass *Challenge* mode. A circle at position $(2, 5)$ for example means that the student answered five questions in a row correctly before making a mistake at question 6, when trying to pass *Challenge* mode for the 2^{nd} time. The size of the circle denotes the number of tug-of-war configurations simulated in *Explore* mode right before the actual *Challenge* attempt. A cross means that the student did not explore at all, but went directly back to *Challenge* mode. The illustration of the explored set-ups (bottom) demonstrates that student B systematically tests the relations between the different characters. Student A and student C on the other hand do not seem to profit from their exploration

0, respectively. The x-axis of Fig. 3 (top) describes the number of challenge attempts: each time the student enters the *Challenge* mode and tries to pass the game is counted as one challenge attempt. The y-axis denotes the level the student achieved in the actual challenge attempt. The level reached is equivalent to the number of questions answered correctly in a row. The size of the circles indicates how many tug-of-war set-ups the student simulated in *Explore* mode. Each time the student simulates a tug-of-war in *Explore* mode, we record the weights (and positions) in the left and right team in a so-called tug-of-war set-up. A cross stands for zero tug-of-war set-ups. Therefore, in the 3^{rd} attempt to pass the *Challenge* mode, student A answered five questions in a row correctly before making a mistake. The student did not explore at all before that attempt, but entered *Challenge* mode directly. The sample trajectories suggest that low performing posttest students need more challenge attempts to pass the game. Indeed, there is a significant negative correlation ($\rho = -0.28$, $p = .001$) between the number of challenge attempts and the achieved posttest score. While all three example students explore several set-ups at the beginning, student A gives up on simulating set-ups soon after the initial simulation phase. This observation suggests that students with a good posttest performance explore more: the average number of

tug-of-war set ups tested in *Explore* mode in-between two attempts to pass the *Challenge* mode is positively correlated to posttest accuracy ($\rho = 0.18$, $p = .048$). Yet, Fig. 3 (top) also shows, that student C keeps exploring, but does not seem to profit from the simulated tug-of-war set-ups.

One hypothesis for the reason behind these observations is that the good performers possess better inquiry strategies, i.e., they simulate more informative tug-of-war set-ups. Figure 3 (bottom) illustrates this behavior. Student B systematically simulates tug-of-war set-ups exploring the relations between the different characters. The student tests first how many small characters are equal to one medium character and then goes on to figure out, how many small characters are equal to one large character. This sequence corresponds to testing the following three rules: R_7, R_8, and R_9 (see Table 1). Student C on the other hand just relies on simulating the exact (or a variation of the) set-up of the last question he answered wrong in the *Challenge* mode. An analysis of the further trajectory confirms that student C indeed always just simulates the set-up of the last wrong challenge question. Student A seems to vary one character at a time. This is a good example of the venerable control of variables strategy. However, the student has created a setup that is too complex to figure out how the variables interact with one another. We also observe, that determining the winner of this set-up requires the application of several rules, i.e., $\mathcal{R}_N = \{R_2, R_8\}$. We therefore propose that the fewer rules needed to determine the winner of the tug-of-war set-up, the more informative the set-up is.

Given this observation, we divide all tug-of-war set-ups tested in the *Explore* mode into three categories: 'strong', 'medium', and 'weak'. This categorization is computed automatically based on the set of rules $\mathcal{R}_N$ necessary to determine the winner of the given tug-of-war configuration. We found that a good exploration strategy focuses on isolating one underlying principle at a time. Therefore, a set-up is considered as 'strong', if the student tests exactly one new rule R_i, i.e., $|\mathcal{R}_N| = 1$ and $R_i \in \mathcal{R}_N$ is seen for the first time. If the rule R_i has been tested or seen previously, the set-up is categorized as being 'medium'. If the set-up tests two rules and one of them is the cancellation rule R_2, i.e., $|\mathcal{R}_N| = 2$ and $R_2 \in \mathcal{R}_N$ the tested configuration is labeled as a 'medium' hypothesis. We assume that the student could still draw conclusions (i.e., find a new rule R_i) by first applying the cancellation rule R_2 (see Table 1) and thus reducing the configuration to a set-up testing exactly one rule. If $|\mathcal{R}_N| = 2 \wedge R_2 \notin \mathcal{R}_N$, the tested set-up is put into the 'weak' category. We also categorize tug-of-war set-ups as being 'weak' hypotheses if they require more than two rules to determine the winning side, i.e., if $|\mathcal{R}_N| > 2$. A set-up testing too many principles at the same time does not allow one to draw conclusions on relations between single characters. An analysis of the training data reveals, that better performers indeed seem to have superior exploration strategies: there is a significant positive correlation between the number of 'strong' tug-of-war set-ups tested and the achieved accuracy in the posttest ($\rho = 0.21$, $p = .019$). Thus, one of the outcomes of this analysis is to identify a previously undocumented inquiry strategy, to which we return in the *Discussion* section.

Probabilistic Models of Performance and Strategies

To investigate the influence of the exploration strategies on the prediction accuracy of a model, we built three different probabilistic graphical models. All three models are based on the approach of BKT and use the set of rules (see Table 1) as an underlying representation of knowledge, i.e., as knowledge components. Similar to BKT, we employ one Hidden Markov model (HMM) per rule. The structure of the graphical model is illustrated in Fig. 4. All three models share the same basic underlying structure. The binary latent variable $K_{R_i,t}$ represents, whether the student has mastered rule R_i at time t. In our case, one time step is equivalent to one tug-of-war configuration simulated in *Explore* mode or one question answered in *Challenge* mode. The input variable $O_{R_i,t}$ is also binary and indicates, whether a student has correctly applied R_i at time t. Prediction is performed as follows: the predicted probability $\hat{p}_{C,t}$ that the student will correctly determine the winning team of a tug-of-war configuration C at time t depends on the predicted probabilities $\hat{p}(O_{R_i,t} = 1)$ of the rules $R_i \in \mathcal{R}_{N_C}$:

$$\hat{p}_{C,t} = \prod_{R_i} \hat{p}(O_{R_i,t} = 1),\ R_i \in \mathcal{R}_{N_C}. \tag{1}$$

We make one small adjustment to the traditional BKT model: we allow the forgetting parameter to be non-zero ($p_F >= 0$). Recent work on comparing different types of knowledge tracing models (Khajah et al. 2016) has demonstrated that including forgetting into BKT leads to superior predictive performance. While all three different probabilistic graphical models share this same basic structure, they differ with respect to their input. As mentioned above, $O_{R_i,t}$ describes, whether a student has applied rule R_i at time t correctly. However, from the students' interaction data, we do not get information about a specific rule R_i, but about the rule set $\mathcal{R}_{N_C}$: in *Challenge* mode, we observe whether the student has answered a question associated with

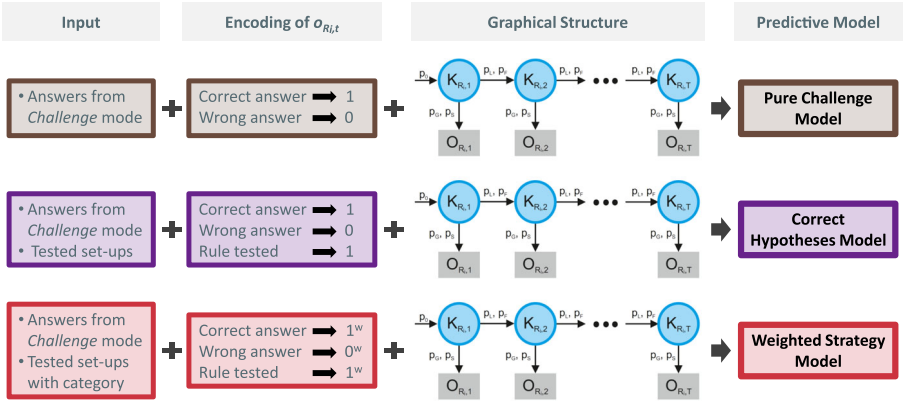

Fig. 4 The Pure Challenge Model (PCM), Correct Hypotheses Model (CHM) and Weighted Strategy Model (WSM) share the same graphical structure and the following parameters: p_0 (initial probability of knowing a rule a-priori), p_L (probability of learning an unknown rule), p_F (probability of forgetting a known rule), p_S (probability of incorrectly applying a known rule), and p_G (probability of correctly applying an unknown rule). However, they differ with respect to their input and the way this input is encoded in terms of the observed variables $\mathbf{o_{R_i}} = [o_{R_i,1}, ..., o_{R_i,T}]$

a tug-of-war configuration C correctly. In *Explore* mode, we record the tug-of-war configuration C that the student has tested. However, most tug-of-war configurations C require more than one rule R_i to determine the winner, i.e. $|\mathcal{R}_{N_C}| > 1$. Therefore, $O_{R_i,t}$ is not directly observed, but derived from the interaction data. The encoding of the variable $O_{R_i,t}$ is then different for each model type.

Pure Challenge Model. The pure challenge model (PCM) encodes correctness as follows: If a student answers a challenge question at time t correctly, we assume that all rules $R_i \in \mathcal{R}_{N_C}$ have been applied correctly, i.e., $o_{R_i,t} = 1, \forall R_i \in \mathcal{R}_N$. If the student gives an incorrect answer, we know that at least one rule $R_i \in \mathcal{R}_N$ has been applied incorrectly. As we cannot directly observe, which rule(s) are unknown to the student, we assume that all rules $R_i \in \mathcal{R}_N$ have been applied incorrectly, i.e., $o_{R_i,t} = 0, \forall R_i \in \mathcal{R}_N$. Note that in the PCM, we do not represent actions performed in the *Explore* mode.

Correct Hypotheses Model. The correct hypotheses model (CHM) is an extension of the PCM. We encode the answers to the challenge questions in the same way as for the PCM. However, in contrast to the PCM, the CHM also incorporates the actions performed in *Explore* mode. For each tug-of-war set-up H tested in *Explore* mode, we first compute the rule set $\mathcal{R}_{N_H}$ necessary to determine the winning side of the simulated tug-of-war configuration. We then assume that all rules in $\mathcal{R}_{N_H}$ have been applied correctly, i.e., $o_{R_i,t} = 1, \forall R_i \in \mathcal{R}_{N_H}$. Let us assume that the student has simulated the set-up illustrated in Fig. 2 at time t. For this set-up, $\mathcal{R}_{N_H} = \{R_2, R_8\}$. We therefore assume that the student has applied R_2 and R_8 correctly at time step t, and feed $o_{R_2,t} = 1$ and $o_{R_8,t} = 1$ into the probabilistic models for rule R_2 and rule R_8.

Weighted Strategy Model. The weighted strategy model (WSM) is based on the observation that exploration behavior significantly influences posttest performance. We encode answers in *Challenge* mode as described in the PCM and rules encountered in *Explore* mode as explained in the CHM. However, the WSM introduces a weighting of the different observations. Observations associated with a tug-of-war set-up simulated in *Explore* mode are weighted according to the three categories 'strong', 'medium', 'weak' as defined previously. Observations associated with answers in *Challenge* mode are weighted based on their correctness. The sequence of T observations $\mathbf{o_{R_i}}$ for a rule R_i is therefore given by

$$\mathbf{o_{R_i}} = (o_{R_{i,1}}^{w_1}, o_{R_{i,2}}^{w_2}, ..., o_{R_{i,T}}^{w_T}), \tag{2}$$

with weights w_j, $j \in 1, ..., T$ specified as follows:

$$w_j = \begin{cases} w_{hs} & o_{R_{i,j}} \text{ is a strong hypothesis.} \\ w_{hm} & o_{R_{i,j}} \text{ is a medium hypothesis.} \\ w_{hw} & o_{R_{i,j}} \text{ is a weak hypothesis.} \\ w_{cw} & o_{R_{i,j}} \text{ is a wrong challenge answer.} \\ w_{cs} & o_{R_{i,j}} \text{ is a correct challenge answer.} \end{cases} \tag{3}$$

The weights w_{hs}, w_{hm}, and w_{hw} are therefore related to the observations in *Explore* mode, while the weights w_{cw} and w_{cs} are related to the answers in *Challenge* mode. Furthermore, as described in Eq. 2, the weights do not have an influence on the structure or the update equations of the model, because they constitute a pure manipulation of the input sequences $\mathbf{o_{R_i}}$. Therefore, we can treat the weights $w = (w_{hs}, w_{hm}, w_{hw}, w_{cw}, w_{cs})$ as hyperparameters and learn them from the collected data using cross validation. The new feature of the WSM in terms of student modeling is that it is able to represent student knowledge and exploration strategies, i.e., the quality of students' tested hypotheses, jointly in one model. In contrast to previous work (González-Brenes et al. 2014) also allowing for the integration of additional features into a BKT model, in the WSM, the strategies directly influence the (hidden) knowledge state. This technique allows us to carry information about students' quality of exploration collected during the game over to the posttest. Furthermore, the chosen approach training weights for the *Explore* and the *Challenge* mode enables us to directly use the trained model to predict students' posttest answers by treating the posttest as a variation of the *Challenge* mode.

Experimental Evaluation

We evaluated the predictive accuracy of our models within the `TugLet` environment as well as on the posttest using the `ED`. We applied a train-test setting, i.e., parameters were fit on the training data set and model performance was evaluated on the test set. Predictive performance was evaluated using the root mean squared error (RMSE) as well as the area under the ROC-curve (AUC). The RMSE is widely used for the evaluation of student models (Pardos and Heffernan 2010; Wang and Beck 2013; Wang and Heffernan 2012; Yudelson et al. 2013). The AUC is a useful additional measure to assess the resolution of a model, i.e., how much predictions of the model differ from the base rate.

Within-Game Prediction. The prediction accuracy of the PCM and the CHM models on the log files collected from `TugLet` was evaluated using student-stratified (i.e., dividing the folds by students) 10-fold cross validation. As the estimation of model performance during parameter tuning leads to a potential bias (Boulesteix and Strobl 2009; Varma and Simon 2006), we use a **nested** 10-fold student-stratified cross validation to estimate the predictive performance of the WSM and at the same time to learn the optimal weights w_{opt} for this model. We learned the parameters $p_i \in \{p_0, p_L, p_F, p_G, p_S\}$ of all the models using a Nelder-Mead (NM) optimization (Nelder and Mead 1965). The NM algorithm is often used for optimization problems due to its simplicity and fast convergence rate. We used $r = 50$ random re-starts for the NM algorithm, because the NM algorithm is known for getting trapped into local optima and to be sensitive to the initial starting values (Nelder and Mead 1965; Parkinson and Hutchinson 1972). We used the same parameter constraints for all models: $p_i \leq 0.5$, if $i \in \{L, F, G, S\}$. The prior probability p_0 remained unconstrained. We optimized the weights for the WSM model using a grid search, i.e., we set $w = (w_{hs}, w_{hm}, w_{hw}, w_{cw}, w_{cs}) \geq 1$ and $w = (w_{hs}, w_{hm}, w_{hw}, w_{cw}, w_{cs}) \leq 4$ and searched all valid combinations.

We then predicted students' answers in *Challenge* mode (correct/wrong) using the trained models and applying Eq. 1.

The WSM demonstrates the highest accuracy within the game, i.e., when predicting the student answer (correct/wrong) to a given *Challenge* question at any point in the game ($\text{RMSE}_{\text{WSM}} = 0.33$). The inclusion of exploration choices into the model led to a reduction in RMSE by 2.6% ($\text{RMSE}_{\text{PCM}} = 0.36$, $\text{RMSE}_{\text{CHM}} = 0.35$), the representation of strategies further reduced the RMSE by 4.4% ($\text{RMSE}_{\text{CHM}} = 0.35$, $\text{RMSE}_{\text{WSM}} = 0.33$). A one-way analysis of variance performed on the per-student RMSE, i.e., the RMSE computed over the predicted *Challenge* answers separately for each student, of the different models shows that there are indeed significant differences between the mean RMSEs of the different models ($F(2, 366) = 6.90$, $p < .01$). Post hoc comparisons using the Tukey HSD test indicate that there is no significant difference in performance between the PCM and the CHM ($p = .34$), while the WSM significantly outperforms the PCM ($p < .001$). The difference between the WSM and the CHM shows a trend to significance ($p = .06$). The differences in per-student AUC, i.e., the AUC again computed separately for each student using his or her predicted *Challenge* answers, between the models are not significant ($\text{AUC}_{\text{PCM}} = 0.80$, $\text{AUC}_{\text{CHM}} = 0.79$, $\text{AUC}_{\text{WSM}} = 0.80$).

The optimal weights found for the WSM are $w_{opt} = \{3, 1, 1, 1, 2\}$. Tug-of-war set-ups classified as 'strong' hypotheses have a higher impact than set-ups falling in the 'medium' or 'weak' categories ($w_{hs} = 3, w_{hm} = 1, w_{hw} = 1$). 'Strong' hypotheses are also assigned more weight than correct answers to challenge questions ($w_{hs} = 3, w_{cs} = 2$). This result demonstrates that the identified inquiry strategy indeed seems to be predictive for student learning.

Posttest Prediction. To evaluate the predictive performance of the different models on the posttest, we used all within-game observations (i.e., actions performed within the `TugLet` environment) for training and predicted the outcome of the external posttest. We again used $r = 50$ random re-starts for the NM algorithm. We constrained the parameters of all models as described for the within-game prediction: $p_i \leq 0.5$, if $i \in \{L, F, G, S\}$. The prior probability p_0 remained unconstrained. For the WSM, we can safely use the optimal weights $w_{opt} = \{3, 1, 1, 1, 2\}$ found in the nested cross validation, since this optimization was performed on within-game data only. We predicted students' answers in the posttest (right/wrong) again by applying Eq. 1. Prediction accuracy in terms of the RMSE and the AUC was computed using bootstrap aggregation with re-sampling ($b = 100$). Figure 5 (left) displays the error measures (with standard deviations) for the PCM, CHM, and WSM models.

The WSM shows the best performance for both error measures. Modeling exploration behavior even in a simplistic way leads to an improvement in RMSE of 4.55% ($\text{RMSE}_{\text{PCM}} = 0.44$, $\text{RMSE}_{\text{CHM}} = 0.42$). Categorization of the different explored set-ups plus the introduction of weighted observations decreases the RMSE by another 7.6% ($\text{RMSE}_{\text{CHM}} = 0.42$, $\text{RMSE}_{\text{WSM}} = 0.39$).

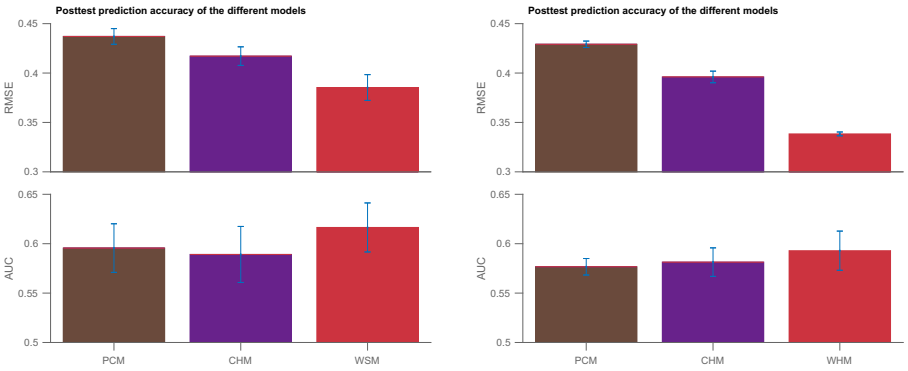

Fig. 5 Comparison of posttest prediction accuracy of the PCM (brown), the CHM (purple), and the WSM (red) on the ED (left) and the VD (right). The WSM outperforms the two other models regarding the RMSE and the AUC on both data sets

The low standard deviations in RMSE ($\sigma_{PCM} = 0.01$, $\sigma_{CHM} = 0.01$, $\sigma_{WSM} = 0.01$) indicate significant differences between the different models. A one-way analysis of variance confirms that there are indeed significant differences between the RMSEs of the different models ($F(2, 297) = 633.46$, $p < .001$). Post hoc comparisons using the Tukey HSD test indicate that all model means are significantly different from each other ($p < .001$ for all comparisons).

The WSM also exhibits a higher AUC than the PCM and the CHM ($\text{AUC}_{\text{PCM}} = 0.60$, $\text{AUC}_{\text{CHM}} = 0.59$, $\text{AUC}_{\text{WSM}} = 0.62$). Although the standard deviations ($\sigma_{PCM} = 0.02$, $\sigma_{CHM} = 0.03$, $\sigma_{WSM} = 0.02$) are higher than for the RMSE, a one-way analysis of variance suggests that the mean AUCs of the different models are not the same ($F(2, 297) = 30.22$, $p < .001$). Post hoc comparisons using the Tukey HSD test indicate that the differences between the PCM and the CHM are not significant ($p = 0.18$), while the WSM significantly outperforms the PCM ($p < 0.001$) and the CHM ($p < 0.01$). We therefore conclude that jointly representing students' accuracy and how students go about learning offers a better representation of student learning than a pure performance model.

Model Validation

To assess the validity of our findings, we applied the PCM, CHM, and WSM to the VD. To compute the prediction accuracy of the PCM, the CHM, and the WSM on the VD, we used the ED as our training data set: we learned the parameters of all three models and the optimal weights for the WSM using all within-game observations (i.e., actions performed within the TugLet environment) from the ED. The VD was used as our test data set: we used the models along with their parameters and weights trained on the ED to predict the outcome of the external posttest of the VD. Figure 5 (right) displays the error measures (with standard deviations) for the PCM, CHM, and WSM models.

As observed already on the ED, the WSM shows the best results for both error measures. Including exploration behavior into the model decreases the RMSE by 7.7% ($RMSE_{PCM} = 0.43$, $RMSE_{CHM} = 0.40$). The weighting of the different strategies leads to a further improvement of 14.6% ($RMSE_{CHM} = 0.40$, $RMSE_{WSM} = 0.34$). A one-way analysis of variance indicates that there are indeed significant differences between the mean RMSEs of the different models ($F(2, 147) = 6473.93$, $p < .001$). Post hoc comparisons using the Tukey HSD test indicate that all the models' mean RMSEs are significantly different from each other ($p < .001$ for all pair-wise comparisons).

The WSM also exhibits a higher AUC than the two other models ($AUC_{PCM} = 0.58$, $AUC_{CHM} = 0.58$, $AUC_{WSM} = 0.59$). Again, a one-way analysis of variance confirms that there are significant differences between the mean AUCs of the three models ($F(2, 147) = 15.64$, $p < .001$). Post hoc comparisons using the Tukey HSD test indicate that while the differences between the PCM and the CHM are not significant ($p = 0.26$), the WSM significantly outperforms the other two models ($p < .001$).

These findings demonstrate that our model is valid for different data sets. Therefore, the identified new inquiry strategy, characterized by the testing of 'strong' set-ups, generally seems to be the optimal strategy for reliably learning the content.

Groups of Children with Similar Inquiry Strategies

In the previous section, we demonstrated that modeling students' inquiry behavior increases our ability to predict transfer: the WSM ties or outperforms the PCM and the CHM when predicting students' answers during the game, and, more importantly, it significantly outperforms the other models when predicting who has truly learned the rules of the game (as assessed by the posttest). The WSM represents student answers and the quality of their inquiry jointly in one model, providing an accurate prediction of students' posttest performance. This result suggests that measuring what students learn in terms of their right/wrong answers is not enough. It is important to also measure how they learn in terms of the quality of their exploration. In order to use our findings for assessment as well as for instruction in terms of a targeted intervention, a more detailed profile of the different choices and strategies is necessary. Ideally, the profiles will be semantically transparent so they can support teacher decision making, though this is not guaranteed by using a clustering algorithm, which we describe next.

Clustering is an approach to identify groups of students with similar characteristics in the data. In contrast to the developed joint models of performance and strategies, which represent the average student behavior (parameters are fit based on the whole data set), clustering allows for describing the behavior of different groups. Given the observed differences between the exploration and challenge behaviors of the students, we were interested in finding groups of children with similar inquiry patterns.

In this section, we demonstrate that we can derive semantically interpretable clusters of students with similar inquiry behavior. One cluster for example contains

students, who systematically test the principles behind the tug-of-war, while another cluster captures students who just try to beat the game. The found clusters are not only correlated with posttest scores, but also with students' science grades and standardized assessments in math. Furthermore, we obtain the same cluster structure on a second independent data set: the high cluster stability (Lange et al. 2004) of 0.82 provides another validation of our cluster solution.

Extracted Features and Clustering Algorithm

Students are clustered after having completed the game using features describing their trajectory through the game, listed in Table 3. We extracted features from the log data, describing three dimensions of inquiry behavior. The first dimension illustrates how fast the students learn and is reflected by the number of challenge questions **NC** needed to pass the game. Our analysis have shown that there is a significant negative correlation between **NC** and the posttest score. The second dimension indicates the student's inquiry behavior, i.e., whether the student tries to figure out the principles of the tug-of-war or just wants to beat the game. We describe this dimension using the number of explored set-ups **NE**. The third dimension indicates whether the student possesses good strategies for inquiry. This dimension is reflected by the number of strong explores **NSE**. The quality of the tested set-ups was found to be predictive for transfer in our previous analysis.

To describe student performance and behavior over time, all the features are calculated by level. We divide the game into eight levels, which is exactly the number of correct challenge answers in a row needed to pass the game. Level n is marked as reached when the student answers exactly n challenge questions in a row correctly for the first time. The features are therefore cumulative and can be described using eight-dimensional vectors. If a level $n-1$ is never reached because a student jumps from level $n-2$ directly to level n, the value from level n is used to fill in position $n-1$.

Therefore,

$$\mathbf{NC_B} = [1, 5, 5, 10, 19, 19, 19, 19] \tag{4}$$

for student B in Fig. 3.

We compute the pair-wise dissimilarities d_{ij} between all students for each feature using the Euclidean distance as our similarity measure between the feature vectors of student i and student j. For n students, we obtain the three $n\ x\ n$ dissimilarity matri-

Table 3 Extracted features and abbreviations (bold) used in the following

Feature	Description
Number of challenge questions	Total number of challenge questions until passing a level.
Number of explored set-ups	Total number of set-ups tested until passing a level.
Number of strong explores	Total number of explored set-ups rated as strong until passing a level.

ces $\mathbf{D_{NC}}$, $\mathbf{D_{NE}}$, $\mathbf{D_{NSE}}$. After normalizing the dissimilarity matrices, they are summed up to the overall dissimilarity matrix $\mathbf{D}$, i.e.,

$$\mathbf{D} = \mathbf{D_{NC}} + \mathbf{D_{NE}} + \mathbf{D_{NSE}}. \tag{5}$$

Since the distance matrix $\mathbf{D}$ describes relations (i.e., distances) between the students, we performed pair-wise clustering (PC) (Hofmann and Buhmann 1997) on $\mathbf{D}$. The PC algorithm takes the distance matrix $\mathbf{D}$ as input and performs a kernel transformation to a (usually higher-dimensional) Euclidean space. In this space, the dissimilarity values can be interpreted as distances between points. As a next step, K-Means Clustering is applied to the Euclidean embedding of the dissimilarity data. The optimal number of clusters k_{opt} is determined by the Bayesian Information Criterion (BIC) using the algorithm presented by Pelleg and Moore (2000): the log-likelihood of the cluster solution is computed under the assumptions that the clusters follow identical spherical Gaussian distributions, which is the type of distribution assumed by the K-Means algorithm. For a solution with k clusters, the free parameters are the $k - 1$ prior probabilities for the clusters, the $m \cdot k$ cluster centroids coordinates (were m is the dimension of the embedding Euclidean space), and one variance estimate.

Resulting Clusters

We clustered the $n_p = 111$ students (out of $n = 127$ students) of the ED, who passed the game. The best BIC score was reached for $k = 7$ clusters. A detailed description of each cluster is given in Table 4: one row corresponds to one cluster and denotes the cluster centroids along with the mean posttest score per cluster and the mean cluster science grade. To keep the table legible, we only display four dimensions of the cluster centroids (levels 1, 3, 5, and 8). In TugLet, challenge questions are ordered by difficulty as follows: one very easy question (level 1), two easy questions (level 3), two medium questions (level 5), and three difficult questions (level 8). To examinate the relation between the clusters and students' grades, we sort the clusters according to the average posttest scores, which were not fed into the clustering algorithm. We

Table 4 Per cluster data for the ED: Cluster centroids (levels 1, 3, 5, and 8) for the features **NC** (Number of Challenge Questions), **NE** (Number of Explored Set-Ups), and **NSE** (Number of Strong Explores), average posttest score per cluster (Posttest), average science grade per cluster (Grade)

NC				NE				NSE				Posttest	Grade
1	3	5	8	1	3	5	8	1	3	5	8		
3.1	5.1	16.9	23.2	6.0	7.9	13.3	14.1	1.0	1.8	4.6	4.7	3.1	0.89
3.8	5.0	16.2	21.8	7.6	8.4	12.6	13.9	0.7	0.9	2.4	2.4	2.6	0.81
4.2	6.0	12.8	28.3	3.2	3.6	4.3	5.7	0.0	0.0	0.0	0.1	2.3	0.81
3.9	4.4	16.6	71.1	7.3	8.0	11.9	23.2	1.0	1.0	3.3	4.3	2.1	0.82
4.2	5.0	15.6	48.4	4.5	4.6	6.1	8.1	0.8	0.8	1.5	1.8	1.8	0.77
4.6	5.5	13.4	15.0	4.1	4.4	6.0	6.2	0.5	0.6	1.5	1.5	1.7	0.78
4.4	5.5	16.3	89.8	4.1	4.7	5.1	7.5	0.6	0.6	0.8	1.2	1.1	0.77

then treat the cluster labels as ordinal variables, allowing us to compute correlations. The posttest score is negatively correlated to the cluster labels ($r = -0.35$, $p < .001$) and the science grade is also negatively correlated to the cluster labels ($r = -0.26$, $p < .01$). These significant correlations indicate that the inquiry behavior represented in the clusters is indeed predictive for the learning outcome and evidently for school grades (due to the limited cluster sizes, we were not able to perform pairwise statistical tests on the differences between the clusters).

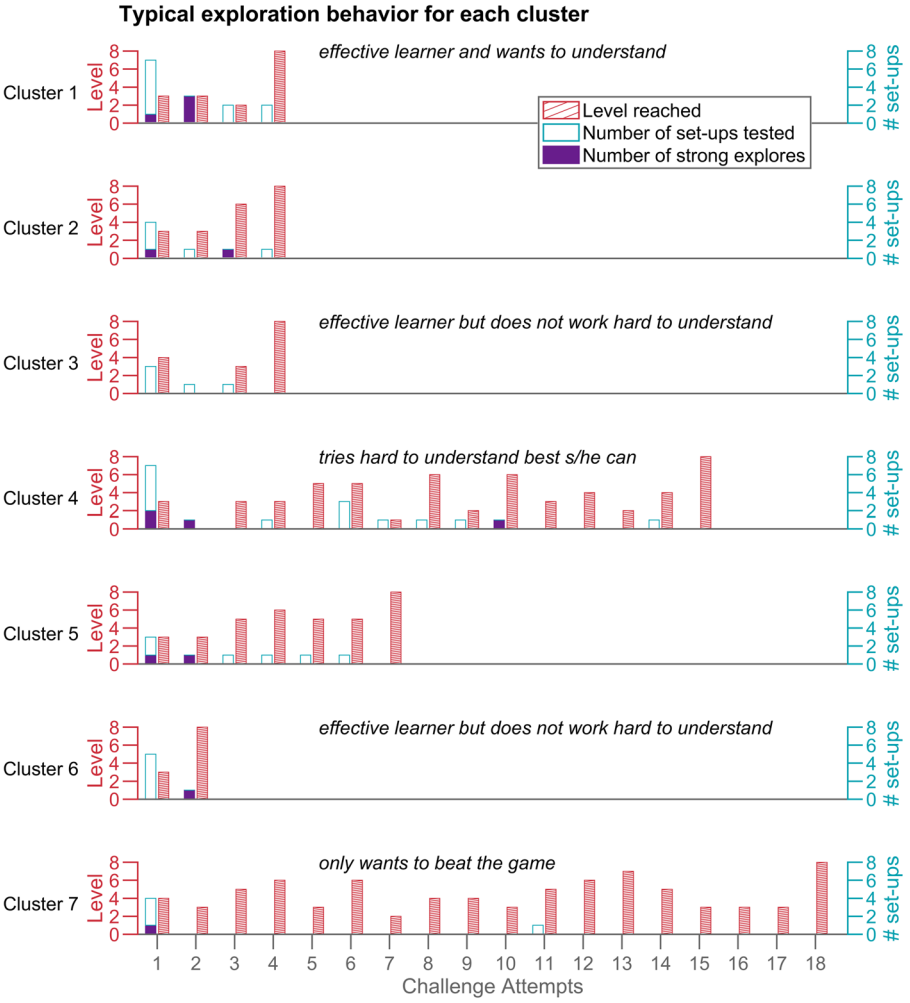

Fig. 6 Typical student for each of the seven clusters. The x-axis denotes the number of challenge attempts. The left y-axis denotes the level (number of correctly answered challenge questions) and the right y-axis denotes the number of set-ups tested. The red hatched bars denote the level reached in the actual challenge attempt. The number of tested set-ups right before entering the challenge mode is indicated by the blue empty bars. The number of simulated configurations which are categorized as 'strong' is displayed with purple filled bars. Clusters 2 and 5 are not assigned to a learning profile as the reflected behaviors seem to lie in-between other clusters

The resulting clusters can be interpreted. Figure 6 illustrates the typical inquiry behavior for each cluster by showing the trajectory through the game for a prototypical student of each cluster. The selected student s_c for each cluster is the student whose exploration behavior is closest to the centroid: for each cluster c and each cluster member s_i, we compute the normalized Euclidean distances $d_{NC,i} = \left\| NC_c - NC_{s_i} \right\|^2$, $d_{NE,i} = \left\| NE_c - NE_{s_i} \right\|^2$, and $d_{NSE,i} = \left\| NSE_c - NSE_{s_i} \right\|^2$. We then get the example student for each cluster by calculating $s_c = argmin_i(d_{NC,i} + d_{NE,i} + d_{NSE,i})$. In the following, we describe the clusters found in detail and provide a provisional semantic interpretation.

Cluster 1: Students in this cluster pass the game with about only five attempts in the *Challenge* mode. They also show the highest average score in the posttest as well as the highest average science grade. The prototypical student from this cluster (see Fig. 6), tests the relations between a large and a small character (rule R_5) as well as position independence in the initial exploration phase. After having failed the *Challenge* mode for the first time, the student systematically tests relations between the different characters by simulating set-ups using rules R_7, R_8, and R_9 (see student B in Fig. 3). The only thing not making this student's inquiry behavior perfect, is, that the student needed to re-test rules R_7 and R_9 before finally passing the game. Based on the described learning characteristics, we assign the following label to this cluster: *effective learner and wants to understand*.

Cluster 2: Students in this cluster also pass the game fast. As can be seen in Fig. 6, the example student from cluster 2 is equally fast at passing the game as the typical student from cluster 1, but simulates less set-ups before each challenge attempt. However, this student exhibits a less systematic inquiry behavior demonstrated by the fewer strong set-ups tested. This cluster cannot be directly assigned to a learner profile as it seems to be in the middle between clusters 1 and 3.

Cluster 3: This cluster consists of students who manage to learn through presented examples, i.e., they are able to efficiently gain the necessary knowledge to pass the game through the *Challenge* mode. Compared to cluster 1, the typical student of this cluster simulates less set-ups and is less systematic, i.e., none of the tested set-ups of the example student is categorized as 'strong' (see Fig. 6). Based on the learning characteristics listed in Table 4, we assign this cluster the following label: *effective learner but does not work hard to understand*.

Cluster 4: Students in this cluster try hard to understand: they test a lot of different set-ups and also achieve a high number of strong set-ups. The trajectory of the sample student (illustrated in Fig. 6) demonstrates that the student overall tested 16 set-ups and even simulated four set-ups, which are rated as 'strong'. We therefore assign this cluster the following label: *tries hard to understand best s/he can*.

Cluster 5: The students in this cluster on average need more time to pass the game than the fast students of the first three clusters. The typical student

(see Fig. 6 for the trajectory) does test (few) tug-of-war configurations (mostly exactly one) in-between challenge attempts. A closer look into the data reveals that the example student does not know how to explore well. Again, we do not assign this cluster to a learner profile as it lies between the clusters 3 and 7.

Cluster 6: Similar to cluster 3, students in this cluster do not explore a lot and beat the game through the *Challenge* mode. As visible from Fig. 6, the typical student of cluster 6 only needs two challenge attempts to pass the game and does not test many different set-ups. The cluster's average posttest score of 1.7 demonstrates, however, that most of these students have not learned the content and probably were lucky at guessing. Therefore, we label them with the same name as cluster 3: *effective learner but does not work hard to understand*.

Cluster 7: The typical student in this cluster needs a long time to pass the game. Figure 6 reveals that the typical student of this cluster does not bother with simulating tug-of-war set-ups, but rather tries to pass the game by using the *Challenge* mode only. The high number of challenge questions answered by students in this cluster ($\mathbf{NC}_8 = 89.9$) and the mean posttest score of 1.1 demonstrate that this strategy is neither efficient nor successful. We assign this cluster the following label: *only wants to beat the game*.

The cluster solution again confirms that not only students' choices between *Challenge* mode and *Explore* mode matter, but it is important how they explore. It also shows that fast learners can (at least in this game) get away with suboptimal learning strategies.

Cluster Validation

To assess the validity of the obtained clusters, we used the same algorithm as before to group the $n_p = 147$ students (out of $n = 152$ students) of the VD, who passed the game. The best BIC score was reached for $k = 6$ clusters. The cluster centroids along with the mean posttest score per cluster, the mean cluster science grade, and the performance in standardized assessments are listed in Table 5. The standardized assessments used are the SBAC math assessment[1] and the SBAC English Language Arts/Literacy assessment[1]. Scores for these assessments fall between 2000 and 3000. A score in the SBAC math assessment bigger than 2652 means that the student exceeded the standard, scores between 2586 and 2652 denote that the standard was met. As we can see from Table 5, only students from cluster 6 (*only wants to beat the game*) did not meet the standard on average. And students belonging to cluster 1 (*effective learner and wants to understand*) tend to exceed the standard. For the SBAC English Language Arts/Literacy assessment students from all clusters met the standard (scores between 2583 and 2681).

[1] http://www.caaspp.org/rsc/pdfs/CAASPP.post-test_guide.2016-17.pdf

Table 5 Per cluster data for the VD: Cluster centroids (levels 1, 3, 5, and 8) for features **NC** (Number of Challenge Questions), **NE** (Number of Explored Set-Ups), and **NSE** (Number of Strong Explores), average posttest score (Posttest), average science grade (Grade), average scores in the SBAC math assessment (SBAC Math) and in the SBAC English Language Arts/Literacy assessment (SBAC Lit)

NC				NE				NSE				Posttest	Grade	SBAC	
1	3	5	8	1	3	5	8	1	3	5	8			Math	Lit
3.6	4.8	16.7	24.6	5.3	5.8	9.9	10.9	1.7	2.0	4.6	4.9	3.1	0.92	2667	2648
4.0	5.0	15.1	18.5	6.6	7.1	8.7	8.8	1.5	1.7	2.7	2.7	2.7	0.91	2664	2620
4.4	5.2	14.6	20.9	3.6	3.7	4.6	5.0	0.6	0.6	1.2	1.2	2.6	0.89	2645	2648
3.9	6.3	17.5	49.9	8.5	9.1	10.4	12.4	1.5	1.6	2.2	2.4	2.5	0.88	2636	2644
3.5	4.9	17.1	75.7	4.5	4.9	6.8	9.8	1.7	1.8	2.9	3.4	2.2	0.9	2653	2612
3.6	5.2	16.2	85.9	1.9	2.2	2.8	4.2	0.8	0.8	1.1	1.2	1.5	0.85	2576	2640

To quantitatively assess the reproducibility of the original clustering solution found on the ED, we compute the clustering stability (Lange et al. 2004) between the new clustering solution found on the VD and the original clustering solution. We use a k-nearest-neighbor classifier trained on the ED, to assign each sample from the VD to a cluster c of the ED, resulting in a vector of predicted labels $\mathbf{l_p}$. The cluster labels $\mathbf{l_{VD}}$ of the clustering solution found on VD serve as ground truth. The cluster stability S is then defined as the normalized Hamming distance between $\mathbf{l_p}$ and $\mathbf{l_{VD}}$. Note that we are comparing two sets of labels that are not necessarily in correspondence. For example, the cluster labeled with 1 in the first solution might correspond to the cluster labeled with 3 in the second solution. Therefore, we optimally permute the label indices of the first solution to maximize the agreement between the two solutions. The cluster stability S is calculated for the permutation with the minimal Hamming distance. For our two clustering solutions, we obtain $S = 0.18$. In other words, the agreement between the two solutions is 82%. The optimal permutation for the label indices found on the original clustering solution is $p_{opt} = \{1, 2, 3, 5, 4, 3, 6\}$.

Cluster 1: Similar to the ED, this cluster consists of students who learn fast and explore strong set-ups. Students in cluster 1 of the VD generally explored less ($\mathbf{NE}_8 = 10.9$) than the students of the ED ($\mathbf{NE}_8 = 14.1$), however, they had an even more efficient inquiry behavior and therefore tested on average the same total number of strong set-ups (ED: $\mathbf{NSE}_8 = 4.7$, VD: $\mathbf{NSE}_8 = 4.9$).

Cluster 2: This cluster consists of students showing good inquiry behavior, but testing less strong set-ups than students in cluster 1, which corresponds to the behavior of students in cluster 2 of the ED.

Cluster 3: The optimal permutation p_{opt} indicates that cluster 3 of the VD is the joint version of clusters 3 and 6 of the ED, which consist of students who quickly pass the game with almost no exploration. This is supported by the fact that the centroid of the joint version of clusters 3 and 6 of the ED ($\mathbf{NC}_8 = 19.6$, $\mathbf{NE}_8 = 6.0$, $\mathbf{NSE}_8 = 1.0$) is very close to the centroid of cluster 3 of the VD ($\mathbf{NC}_8 = 20.9$, $\mathbf{NE}_8 = 5.0$, $\mathbf{NSE}_8 = 1.2$).

Cluster 4: Students in this cluster need more time to pass the game than the students in the first three clusters. Similar to students in cluster 3 they try to beat the game in the *Challenge* mode. This cluster corresponds to cluster 5 of the ED.

Cluster 5: This cluster corresponds to cluster 4 from the ED: cluster members need to answer a lot of challenge questions before passing the game, while also exploring strong set-ups. However, while students in cluster 5 of the VD explored an average total number of set-ups ($\mathbf{NE}_8 = 9.8$), students in cluster 4 of the ED tended to extensively simulate different set-ups ($\mathbf{NE}_8 = 23.2$).

Cluster 6: Students in this cluster try to pass the game with repeated attempts in *Challenge* mode. They need a long time to pass the game. Cluster 6 corresponds to cluster 7 of the ED.

We again compute the correlations between academic performance by sorting the cluster labels according to the average posttest scores and treating them as ordinal variables. Similar to the ED, there is a significant correlation between the science grade and the cluster label ($r = -0.2$, $p = .036$). Furthermore, performance in the SBAC math assessment is also significantly correlated to the cluster label ($r = -0.26$, $p < .01$). Note that both of these correlations have been computed based on the optimal permutation of the cluster labels.

While we can find the same types of learners in the VD as in the ED, the share of the different clusters varies over the two data sets. Figure 7 illustrates the cluster distributions of the ED and the VD. Note that we use the optimally permuted cluster labels from the ED for both data sets. The biggest difference lies in cluster 1, which is the cluster containing the top students. 19% of the students from the VD can be classified as fast learners, who systematically test the relations between the different characters. Only 10% of the students of the ED belong to this group. Also the amount of students trying to pass the game through the *Challenge* mode with little success, i.e., needing lots of attempts, is slightly higher for the ED than for the VD (Cluster 7 (ED): 12%, Cluster 7 (VD): 9%). These findings are in line with the fact, that students

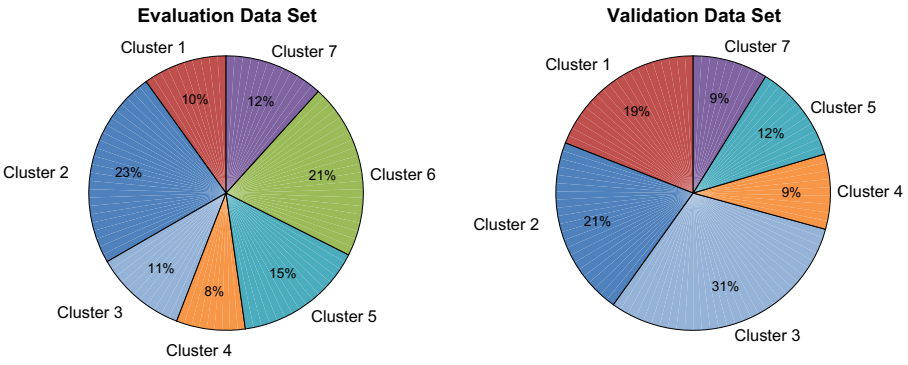

Fig. 7 Distribution of clusters for the ED and the VD. For both data sets, the optimally permuted cluster labels from the ED are used

from the VD performed significantly better on the posttest than students from the ED. Both data sets contain about 30% of students, who manage to pass the game fast using only the *Challenge* mode (Cluster 3 + Cluster 6 (ED): 32%, Cluster 3 (VD): 31%). Also the share of students, who try hard to understand (with little success) is similar for both data sets (Cluster 4 (ED): 8%, Cluster 4 (VD): 9%).

The different shares of students with the optimal learning strategy (i.e., cluster 1) in the two data sets suggests that TugLet along with the clustering algorithm might be used as a formative assessment for teachers to guide instructional decision-making.

Discussion & Conclusion

An important goal of education is to ensure that students gain proficient knowledge of the subject matter, whether it be reading, math, or science. This goal is reflected in assessments and intelligent computer environments that emphasize student accuracy in solving pre-defined problems. A second important goal of education is to prepare students to continue learning on their own by imbuing them with dispositions and strategies for learning. Historically, this second goal has been difficult to assess, other than using surveys. The introduction of highly-interactive computer environments that log copious amounts of user data present the opportunity to capture behavioral data relevant to student dispositions and learning strategies. For example, it has been possible to detect students' use of critical thinking (Chi et al. 2014), literature inquiry (Chin et al. 2016), and feedback seeking behavior (Cutumisu et al. 2015). Moreover, it has become possible to show empirically that some learning strategies are better for learning than other strategies (Gaševic et al. 2017), whereas before, claims of strategy benefits were based more on rational analysis than hard data.

In this paper, we have developed a model able to jointly represent student knowledge and exploration strategies. Our work is comparable to research on engagement modeling, where student knowledge and engagement are simultaneously traced (Schultz and Arroyo 2014). FAST (González-Brenes et al. 2014) also allows for the integration of additional features into a BKT model, however, these additional features influence prediction of the observed state only. In contrast to this approach, in our joint model of knowledge and strategy, the strategies directly influence the (hidden) knowledge state.

Our results demonstrate that even simple probabilistic models of strategies offer a better representation of learning than a pure performance model. Within the game, our modeling approach improves prediction accuracy on the original data set ED by 8.3% (RMSE_{PCM}: 0.36, RMSE_{WSM}: 0.33). In comparison, previous research aiming at improving the prediction accuracy of simple probabilistic models achieved similar or worse results. Individualized BKT (Yudelson et al. 2013) for example reached improvements of RMSE of around 1% (e.g., for an algebra data set: $\text{RMSE}_{best} = 0.363$, $\text{RMSE}_{worst} = 0.361$). Clustered knowledge tracing (Pardos et al. 2012) achieved similar results with improvements in RMSE again around 1%. While these improvements might seem small, it has been shown that even small changes in prediction accuracy have an impact on overpractice and underpractice (Yudelson et al.

2013). For the prediction of the posttest, our combined model of knowledge and strategy yields substantially larger improvements. Modeling the strength of student hypotheses improves the posttest prediction accuracy in terms of RMSE by 11.8% for the ED ($RMSE_{PCM} = 0.44$, $RMSE_{WSM} = 0.39$) and by 20.9% for the validation data set VD ($RMSE_{PCM} = 0.43$, $RMSE_{WSM} = 0.33$). This increase in prediction accuracy demonstrates that including strategies into a model is especially important when predicting overall student learning from an instructional technology and not just performance within the environment. The replication of our results on a different independent data set demonstrates that our findings hold for students from different academic contexts. The students from the VD significantly outperformed the students from the ED on the posttest. Despite these performance differences, incorporating the inquiry strategy found on the ED into the model, improved predictive performance.

Our findings indicate that students' inquiry behavior indeed influences the learning outcome.

Furthermore, we have demonstrated that we can group the children into clusters with similar learning strategies. The clusters found on the ED can be interpreted and the labels found (*effective learner and wants to understand*, *tries hard to understand best s/he can*, *effective learner but does not work hard to understand*, *only wants to beat the game*) can be communicated to teachers. A limitation of this work is that we were not able to assign all the clusters to a learning profile. Our identified learner profiles cover about 2/3 of the children from the ED and the VD. While the rest of the children cannot directly be assigned to a learner profile, their cluster characteristics can still be semantically interpreted. We validated our clustering solution on the VD and found a cluster agreement of 82%: we find the same clusters on the VD, only the distribution of clusters is different. The students of the VD performed better in the posttest. This fact is reflected in the cluster size of the best cluster: 19% of the students from the VD can be labeled as *effective learner and wants to understand*, while only 10% of the students from the ED belong to this learning profile. Thus, the algorithm can conceivably help teachers determine how their class as a whole is performing, relative to other classes. The learning patterns found are not only correlated to students' posttest scores, they seem to be predictive for students' academic achievement in general: the cluster labels are correlated to students' science grades as well as their performance in a standardized math assessment. The game together with the cluster solution can therefore be used as an assessment indicating students' learning strategies, in our case their inquiry strategies, important for future learning.

We also identified a new (positive) inquiry strategy, which was not described in the literature before. We categorized simple tug-of-war set-ups t esting exactly one rule at a time as 'strong'. We call this new inquiry strategy: *keep it simple to isolate equivalence of units*. This strategy is related to the known strategy of CVS (control of variables). However, there is one key difference, which has proven essential for learning and this is the simplicity requirement. CVS can be applied also when doing complex simulations and in this case is not of much help for learning. For example, a student can include a confusing number of causal contributors that make it difficult to infer underlying rules, even if (s)he only changes one of the contributors at a time (e.g., three big, five medium, one small character). While the found strategy might seem obvious for the presented game, our analysis showed that only a fraction of the

students used it. Namely the students assigned to the best performing cluster on both data sets. Therefore, the identified inquiry strategy is able to separate the top students from the rest. Incorporating this inquiry strategy into a probabilistic model improved prediction of transfer on two different data sets. The identified inquiry strategy can generally be considered as good for learning in this environment and we speculate it is useful in other domains as well. For example, in simulations that ask students to learn by creating circuits to make bulbs light (etc.,), it may be profitable to isolate the relation between a battery and a resistor, rather than putting in lots of battery and resistors to make the lights come on at a specific brightness.

To conclude, in this paper we aimed at using artificial intelligence to pave the way for eventually augmenting teachers' abilities to identify students' learning strategies in open-ended exploration environments, and conceivably empower the computer to detect and address learning strategies in these environments. Specifically, we were interested in answering the following three research questions. 1) Can the computer help detect learning strategies? 2) Can we determine which learning strategies are indeed good for learning? 3) Can we characterize them transparently so that a teacher could conceivably use them as formative assessments?

In both parts of the paper, we have semi-automatically detected strategies leading to successful learning. In part 1 of the paper, we found that the quality of students' inquiry is essential, i.e., the quality of tested tug-of-war set-ups (categorized in terms of strong/weak/medium) is what differentiates high-performing from low-performing students. In part 2 of the paper, we used unsupervised clustering and determined the optimal number of clusters in a data-driven way to find groups of students with similar exploration patterns. Both parts were based on human-engineered features. We conclude that we can answer research question 1) with yes, at least in the context of `TugLet`, but hopefully in other environments as well.

Research question 2) can be confirmed, too. Our analysis using probabilistic models jointly representing student performance and exploration strategies demonstrated that including the quality of the tested tug-of-war set-up and hence students' inquiry behavior into the model improves prediction of learning outside the game. This finding indicates that it matters not only what we learn, but also how we learn. While this may seem intuitively obvious, it may be useful to recognize that the behaviorist tradition long held the position that associating a reinforcer with a correct behavior was the key to learning, not the processes – the how – that led to the correct behavior, and certainly not a learning process where learner-chosen strategies play a role (Skinner 1986). Indeed, current knowledge tracing models benefit from the behaviorist tradition by focusing on answer accuracy, whereas student chosen strategies for learning do not enter those models. These are very powerful traditions, but one of the limitations of these traditions is that they have been poor at predicting or supporting transfer (Bransford and Schwartz 1999) which is exactly what we have showed here. The value of representing how students went about learning appeared when students were asked to solve novel problems that depended a near transfer.

Furthermore, we have also shown that the learning strategies found are correlated to science grades and standardized assessments. A next step is to determine the best way to have the computer (or teacher) help the students learn effective inquiry strategies, once the computer detects the strategies they are using.

For research question 3), we have provided the first step in this paper. Our cluster solution can be semantically interpreted and has already been validated on a second data set. As a next step, we plan to further validate the cluster labels by collecting information about teachers' teaching beliefs and teaching style as well as their assessment of students' inquiry behavior. Furthermore, we plan to collect a large-scale data set from a different cultural context to investigate the cultural (and teaching) differences of inquiry strategies. The fact that the clustering yielded similar results across samples of different backgrounds indicates that it may be generally useful in helping teachers gain insight into their students' strategies as well as whether these strategies are successful or not. The final goal is to use `TugLet` as an assessment tool for teachers, which, in the optimal case, provides them with new information about the learning behavior of their students.

Acknowledgments We gratefully thank the support of the Gordon and Betty Moore Foundation, the Marianne and Marcus Wallenberg Foundation, and the National Science Foundation, DRL #1020362.

References

Amershi, S., & Conati, C. (2009). Combining unsupervised and supervised classification to build user models for exploratory learning environments. Journal of Educational Data Mining, pp. 18–71.

Baker, R.S., Corbett, A.T., Koedinger, K.R. (2004). Detecting student misuse of intelligent tutoring systems. In *Proc. ITS* (pp. 531–540).

Baker, R.S.J.d., Corbett, A.T., Roll, I., Koedinger, K.R. (2008). Developing a generalizable detector of when students game the system. *UMUAI*, *18*(3), 287–314.

Barata, G., Gama, S., Jorge, J., Goncalves, D. (2016). Early prediction of student profiles based on performance and gaming preferences. *IEEE Transactions on Learning Technologies*, *9*(3), 272–284.

Beck, J.E., Chang, K.-M., Mostow, J., Corbett, A. (2008). Does help help? introducing the bayesian evaluation and assessment methodology. In *Proc. ITS* (pp. 383–394).

Boulesteix, A.-L., & Strobl, C. (2009). Optimal classifier selection and negative bias in error rate estimation: an empirical study on high-dimensional prediction. *BMC Medical Research Methodology*, *9*(1), 85+.

Bransford, J.D., & Schwartz, D.L. (1999). Rethinking transfer: a simple proposal with multiple implications. *Review of Research in Education*, *24*, 61–100.

Cen, H., Koedinger, K.R., Junker, B. (2007). Is over practice necessary? -Improving learning efficiency with the cognitive tutor through educational data mining. In *Proc. AIED* (pp. 511–518).

Cen, H., Koedinger, K.R., Junker, B. (2008). Comparing two IRT models for conjunctive skills. In *Proc. ITS* (pp. 796–798).

Chi, M., Schwartz, D.L., Blair, K.P., Chin, D.B. (2014). Choice-based assessment: Can Choices Made in Digital Games Predict 6th-Grade Students', Math Test Scores? In *Proc. EDM* (pp. 36–43).

Chin, D.B., Blair, K.P., Schwartz, D.L. (2016). Got game? a choice-based learning assessment of data literacy and visualization skills. *Technology, Knowledge, and Learning*, *21*, 195–210.

Corbett, A.T., & Anderson, J.R. (1994). Knowledge tracing: modeling the acquisition of procedural knowledge. *UMUAI*, *4*(4), 253–278.

Cutumisu, M., Blair, K.P., Chin, D.B., Schwartz, D.L. (2015). Posterlet: a Game-Based assessment of children's choices to seek feedback and to revise. *Journal of Learning Analytics*, *2*(1), 49–71.

Eagle, M., & Barnes, T. (2014). Exploring differences in problem solving with data-driven approach maps. In *Proc. EDM* (pp. 76–83).

Fang, Y., Shubeck, K., Lippert, A., Cheng, Q., Shi, G., Geng, S., Gatewood, J., Chen, S., Zhiqiang, C., Pavlik, P., Frijters, J., Greenberg, D., Graesser, A. (2018). Clustering the learning patterns of adults with low literacy skills interacting with an intelligent tutoring system. In *Proc. EDM* (pp. 348–384).

Fratamico, L., Conati, C., Kardan, S., Roll, I. (2017). Applying a framework for student modeling in exploratory learning environments: comparing data representation granularity to handle environment complexity. *International Journal of Artificial Intelligence in Education*, *27*(2), 320–352.

Gaševic, D., Jovanovic, J., Pardo, A., Dawson, S. (2017). Detecting learning strategies with analytics: links with self-reported measures and academic performance. *Journal of Learning Analytics*, *4*(2), 113–128.

Geigle, C., & Zhai, C. (2017). Modeling MOOC student behavior with two-layer hidden markov models. In *Proc. L@S* (pp. 205–208).

González-Brenes, J.P., Huang, Y., Brusilovsky, P. (2014). General features in knowledge tracing: Applications to multiple subskills, temporal item response theory, and expert knowledge. In *Proc. EDM* (pp. 84–91).

González-Brenes, J.P., & Mostow, J. (2012). Topical Hidden Markov Models for Skill Discovery in Tutorial Data. NIPS - Workshop on Personalizing Education With Machine Learning.

Hofmann, T., & Buhmann, J.M. (1997). Pairwise data clustering by deterministic annealing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *19*(1), 1–14.

Johns, J., & Woolf, B. (2006). A dynamic mixture model to detect student motivation and proficiency. In *Proc. AAAI* (pp. 163–168).

Kardan, S., & Conati, C. (2011). A framework for capturing distinguishing user interaction behaviours in novel interfaces. In *Proc. EDM* (pp. 159–168).

Käser, T., Busetto, A.G., Solenthaler, B., Kohn, J., von Aster, M., Gross, M. (2013). Cluster-based prediction of mathematical learning patterns. In *Proc. AIED* (pp. 389–399).

Käser, T., Hallinen, N.R., Schwartz, D.L. (2017). Modeling exploration strategies to predict student performance within a learning environment and beyond. In *Proc. LAK* (pp. 31–40).

Käser, T., Klingler, S., Schwing, A.G., Gross, M. (2014). Beyond knowledge tracing: modeling skill topologies with bayesian networks. In *Proc. ITS* (pp. 188–198).

Khajah, M., Lindsey, R.V., Mozer, M.C. (2016). How deep is knowledge tracing? In *Proc. EDM* (pp. 94–101).

Kinnebrew, J.S., Loretz, K.M., Biswas, G. (2013). A contextualized, differential sequence mining method to derive students' learning behavior patterns. Journal of Educational Data Mining 5(1).

Lange, T., Roth, V., Braun, M.L., Buhmann, J.M. (2004). Stability-based validation of clustering solutions. *Neural Computation*, *16*(6), 1299–1323.

Mayer, R.E. (2004). Should there be a three-strikes rule against pure discovery learning? the case for guided methods of instruction. American Psychologist, pp 14–19.

Mojarad, S., Essa, A., Mojarad, S., Baker, R.S. (2018). Data-Driven Learner profiling based on clustering student behaviors: learning consistency, pace and effort. In *Proc. ITS* (pp. 130–139).

Nelder, J.A., & Mead, R. (1965). A simplex method for function minimization. *The Computer Journal*, *7*(4), 308–313.

Pardos, Z.A., & Heffernan, N.T. (2010). Modeling individualization in a bayesian networks implementation of knowledge tracing. In *Proc. UMAP* (pp. 255–266).

Pardos, Z.A., Trivedi, S., Heffernan, N.T., Sárközy, G.N. (2012). Clustered knowledge tracing. In *Proc. ITS* (pp. 405–410).

Parkinson, J.M., & Hutchinson, D. (1972). An investigation into the efficiency of variants on the simplex method. In *Numerical Methods for Non-Linear Optimization* (pp. 115–135): Academic Press.

Pavlik, P.I., Cen, H., Koedinger, K.R. (2009). Performance factors analysis - a new alternative to knowledge tracing. In *Proc. AIED* (pp. 531–538).

Pelleg, D., & Moore, A. (2000). X-means: Extending k-means with efficient estimation of the number of clusters. In *Proc. ICML* (pp. 727–734).

Roll, I., Aleven, V., McLaren, B., Koedinger, K. (2011). Improving students' help-seeking skills using metacognitive feedback in an intelligent tutoring system. *Learning and Instruction*, *21*, 267–280.

Roll, I., Baker, R., Aleven, V., Koedinger, K.R. (2014). On the Benefits of Seeking (and Avoiding) Help in Online Problem-Solving Environments. *Journal of the Learning Sciences*, *23*(4), 537–560.

Rowe, E., Baker, R., Asbell-Clarke, J., Kasman, E., Hawkins, W. (2014). Building automated detectors of gameplay strategies to measure implicit science learning. In *Proc. EDM* (pp. 337–338).

Rowe, J., Mott, B., Mcquiggan, W.S., Sabourin, J., LEE, S., Lester, C.J. (2009). Crystal island: a Narrative-Centered learning environment for eighth grade microbiology. In *Proc. AIED Workshops* (pp. 11–20).

Sabourin, J.L., Shores, L.R., Mott, B.W., Lester, J.C. (2013). Understanding and predicting student self-regulated learning strategies in game-based learning environments. *International Journal of Artificial Intelligence in Education*, *23*(1), 94–114.

Sawyer, R., Rowe, J., Azevedo, R., Lester, J. (2018). Filtered time series analyses of student problem-solving behaviors in game-based learning. In *Proc. EDM* (pp. 229–238).

Schultz, S.E., & Arroyo, I. (2014). Tracing knowledge and engagement in parallel in an intelligent tutoring system. In *Proc. EDM* (pp. 312–315).

Schwartz, D.L., & Arena, D. (2013). Measuring what matters most: Choice-based assessments for the digital age. The MIT Press.

Schwartz, D.L., Chase, C.C., Oppezzo, M.A., Chin, D.B. (2011). Practicing versus inventing with contrasting cases: the effects of telling first on learning and transfer. *Journal of Educational Psychology*, *103*(4), 759–775.

Shute, V.J., & Glaser, R. (1990). A large-scale evaluation of an intelligent discovery world: Smithtown. *Interactive Learning Environments*, *1*, 51–77.

Skinner, B.F. (1986). Programmed instruction revisited. *The Phi Delta Kappan*, *68*(2), 103–110.

Truong-Sinh, A., Krauss, C., Merceron, A. (2017). Can Typical Behaviors Identified in MOOCs be Discovered in Other Courses? In *Proc. EDM* (pp. 220–225).

Varma, S., & Simon, R. (2006). Bias in error estimation when using cross-validation for model selection. *BMC Bioinformatics*, *7*(1), 91.

Wang, Y., & Beck, J. (2013). Class vs. student in a bayesian network student model. In *Proc. AIED* (pp. 151–160).

Wang, Y., & Heffernan, N.T. (2012). The student skill model. In *Proc. ITS* (pp. 399–404).

Wieman, C.E., Adams, W.K., Perkins, K.K. (2008). PhET: simulations that enhance learning. *Science*, *322*(5902), 682–683.

Yudelson, M.V., Koedinger, K.R., Gordon, G.J. (2013). Individualized bayesian knowledge tracing models. In *Proc. AIED* (pp. 171–180).

Zhang, N., Biswas, G., Dong, Y. (2017). Characterizing students' learning behaviors using unsupervised learning methods. In *Proc. AIED* (pp. 430–441).