Wheel-Spinning: Students Who Fail to Master a Skill

Joseph E. Beck and Yue Gong

Worcester Polytechnic Institute
{josephbeck,ygong}@wpi.edu

Abstract. The concept of mastery learning is powerful: rather than a fixed number of practices, students continue to practice a skill until they have mastered it. However, an implicit assumption in this formulation is that students are capable of mastering the skill. Such an assumption is crucial in computer tutors, as their repertoire of teaching actions may not be as effective as commonly believed. What if a student lacks sufficient knowledge to solve problems involving the skill, and the computer tutor is not capable of providing sufficient instruction? This paper introduces the concept of "wheel-spinning;" that is, students who do not succeed in mastering a skill in a timely manner. We show that if a student does not master a skill in ASSISTments or the Cognitive Tutor quickly, the student is likely to struggle and will probably never master the skill. We discuss connections between such lack of learning and negative student behaviors such as gaming and disengagement, and discuss alterations to ITS design to overcome this issue.

Keywords: mastery learning, student modeling, wheel-spinning.

1 Introduction

Intelligent Tutoring Systems (ITS) are generally effective learning environments for computer-assisted problem solving. ITS are capable of providing assistance to students who are stuck with problem solving, and have been found to be better than traditional paper and pencil homework for helping students learn. Compared to more traditional methods of instruction, ITS typically perform much better on experimenter-defined measures [e.g., 1], and somewhat better on standardized instruments. Although it is tempting to assume all students benefit from using an ITS, this assumption does not necessarily hold as an ITS is not a strong choice for all learners.

In the mastery learning [2] framework, as implemented in many ITS, the student does not see a fixed number of problems, but continues to solve problems until he achieves mastery of the associated skills. In other words, once the student finishes solving a problem, possibly with the assistance of the computer, if he has not yet mastered the related skill, the computer presents another problem. There has been a long history of work in on mastery learning with computer-based education [3], and this model makes intuitive sense and certainly realizes the maxim of "practice makes perfect," particularly as most tutors provide assistance to the student in the form of hints or breaking the problem into steps. However, a bit of thought reveals some hidden weaknesses in the model. If a student requires assistance to solve the first two

K. Yacef et al. (Eds.): AIED 2013, LNAI 7926, pp. 431–440, 2013.

problems, presenting a third with the hope the student will learn the skill could very well be a sensible strategy. If the student has been unable to solve twenty problems, and required considerable help on all of them, it is probably rather optimistic to believe that the twenty-first problem will enable the student to suddenly acquire the skill (in the data set we analyze, there is only a 1.4% chance such a student will ever master the skill, at least within the data collected for this study).

The assumption that students will eventually acquire skills with enough practice is not just part of tutorial decision making, as knowledge tracing [4] assumes a constant probability of learning the skill on every problem-solving attempt. However, not all students are able to acquire skills within an ITS, and some spend a considerable amount of time stuck in the mastery learning loop without any learning occurring. Aside from simply wasting the learner's time, such an experience is presumably frustrating as learners are repeatedly presented with problems they are clearly unable to solve. We refer to this phenomenon as "wheel-spinning," referring to a car stuck in mud or snow; its wheels are spinning rapidly, but it is not going anywhere. Similarly, students are being presented with many problems, but are not making progress towards mastery. Later, we will discuss possible connections with other negative behaviors such as gaming.

2 Describing Wheel-Spinning

We define wheel-spinning as a student who spends too much time struggling to learn a topic without achieving mastery. Some students will begin working with an ITS already understanding the material. Other students will master the skill relatively quickly, perhaps with the assistance of the ITS's coaching. Neither group is problematic. We are concerned with students who spend too much time without mastering the skill. This definition has two concepts that must be operationalized:

1. What does it mean to *master* a topic?
2. How much time is *too much*?

The answer to both of these questions will vary somewhat by system, as the idea of mastery is a vague concept and can be instantiated in a variety of ways. One approach, proposed by Corbett and Anderson (1995), was to estimate the student's knowledge, and when the probability a student knew a skill exceeds 0.95, then the student is considered to have mastered the skill. An approach used in the ASSISTments system is to consider a student to have mastered the skill upon getting three questions in a row correct. With respect to time, an ideal amount will also vary by system. An ITS whose problems require 10 minutes to solve should probably require fewer problems for mastery than one that requires 20 seconds per problem.

For our mastery criterion, we decided to use the simpler approach of three correct responses in a row. The knowledge tracing model-fitting process is rather slow on large data sets, and there is concern about its ability to disambiguate student knowledge due to issues of identifiability [5]. Also, if others wish to replicate our work on other data sets, a mastery criterion that does not require subscribing to a particular student modeling framework will be easier to work with. We also assume that once students master a skill, they are unable to unmaster it. To be clear, we believe that

forgetting does exist, and a real-world adaptive system needs to account for it. However, our goal here is to understand how learners perform during initial mastery, and whether they are able to achieve such in a reasonable amount of time. Forgetting what was learned, while an important topic, is not central to this research question.

For how much time is a reasonable amount to master, we selected 10 practice opportunities. Although this cutpoint is somewhat arbitrary, we will see (see Fig. 1) that the results are not that sensitive to the exact threshold selected. Furthermore, one of the systems we are analyzing, ASSISTments, has a feature which "locks out" learners after they have made 10 attempts at a skill in a single day, and requires them to try again on a later day. We are not sure what impact this feature could have on the data, or what students might be doing after being locked out (e.g., asking someone for help). Therefore, we used 10 practice opportunities as a threshold for mastering in a reasonable time frame. See Fig. 1 for a visual representation of wheel-spinning behavior in the Cognitive Algebra Tutor (CAT) and in ASSISTments.

In Fig. 1, the x-axis represents how many practice opportunities a student has had on a particular skill, and the y-axis represents the cumulative probability a student has mastered, i.e. gotten three problems in a row correct, the skill. By definition, no student has mastered a skill on the first two practice attempts. On the third practice attempt, approximately 35% of students in both the CAT and ASSISTments have mastered the skill. To achieve mastery this quickly, these students made no mistakes on their first three problems; therefore, these students did not benefit from any of coaching available on this skill. In other words, these students answered the questions without requiring assistance, and were essentially using the tutor as fancy paper and pencil homework; therefore, the ITS should not receive credit for having helped these students.

After three practice opportunities, both CAT and ASSISTments show a gradual rise in the percentage of students having mastered the skill. After 6 practice opportunities, 59% of students in the cognitive tutor and 55% of students in ASSISTments have mastered a skill. Finally, after 10 practice opportunities 69% of students in CAT and 62% of students in ASSISTments have mastered the skill. Although we selected 10 practice opportunities somewhat arbitrarily, and based on a possible artifact in the ASSISTments data set, this threshold is past the "elbow" of the mastery curves for both systems, and inspecting Fig. 1demonstrates that the proportion of students having mastered the skill would not change noticeably if the threshold were increased beyond 10. Therefore, we are satisfied with our threshold for wheel-spinning, at least for a first analysis of this problem.

It is interesting to compare the curves for the CAT and ASSISTments. Although, initially, both tutors had approximately equal numbers of students who already knew the skill, one possible implication is that the CAT is doing a better job of helping students achieve mastery than ASSISTments. Such comparisons should be made with caution, as there are a variety of factors that could influence differences in the curves between systems. First, problems could vary in difficulty. A system with relatively easier problems would show more students achieving mastery than one with harder questions. Making problems easier is probably not a good way of reducing wheel-spinning. However, since both systems have approximately 34% of students immediately mastering the skill, easier problems is an unlikely explanation. Second, patterns of usage can differ. For example, if students solve problems in a particular

skill in large batches in ASSISTments, but only a few per day in CAT, students will have more opportunities for learning outside of the tutor. Thus, we cannot conclude that CAT is better than ASSISTments at providing assistance that prevents wheel-spinning, but can conclude that some combination of CAT and how it is deployed appears to work better than ASSISTments. One stark conclusion from the graph, however, is that a substantial number of students have problems with wheel-spinning, and this issue is not particular to one ITS, as it occurs in two widely-used computer tutors with differing pedagogical approaches.

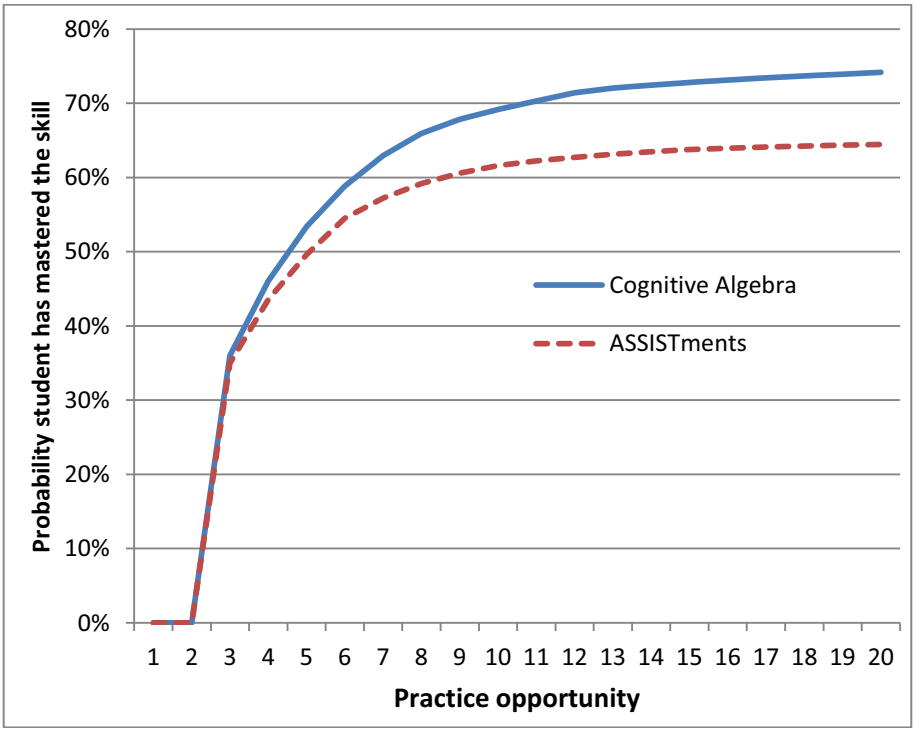

Fig. 1. Graph of wheel-spinning in ASSISTments and Cognitive Algebra Tutor

3 Modeling Wheel-Spinning

Given that wheel-spinning is at best non-productive and possibly irritating for the student, we would like to detect this behavior as rapidly as possible. If we can predict that a student is likely to spin his wheels, we can perform some other tutorial action that is more instructional in nature.

Our approach is to consider each student-skill pair, and look at cases where the student either masters the skill, or after seeing 10 problems has failed to master it (wheel-spinning). Data after the tenth encounter or after the student has mastered the skill are ignored. This definition has an asymmetry, as a student who only sees 7

problems but fails to master the skill, is of indeterminate wheel-spinning, and is not included in this analysis; whereas a student who did master it in 7 attempts is included. Thus, this approach undercounts wheel-spinning, and estimates it occurs in 9.8% of the data. We construct a logistic regression model to predict which category the student will master the skill or wheel-spin. In order to determine how quickly we can categorize students, we build a separate model for each number of practice opportunities the student has had on the current skill. In other words, we construct a model for when the student begins practicing the skill (has seen 0 problems), when he has seen 1 problem, 2 problems, etc. We take this approach for two reasons. First, we want to see how accuracy changes as we accumulate more data about the student. Second, what is important could change over time. Requesting a bottom-out hint (the answer to the problem) on the first item might not be problematic, but requesting such assistance on the fourth problem could be a strong negative indicator. We had 258,990 problems solved by 5997 students. After removing indeterminate data, our data set consists of 131,909 problems solved by 5026 students. This analysis used data collected between September 2010 and July 2011, with students primarily from the northeast United States. We only have student self-reported ages, and 75% of the students asserted they were 12 to 15 years of age on January 1, 2011.

The dependent variable is whether or not a student will wheel-spin on this skill. The first three independent variables track student performance on the skill in question, and the next three look at his performance across all skills:

- Prior number of correct responses by the student on this skill
- Response times on this skill. We first transform response times for each item into a Z score for that item (to account for some problems taking longer than others). We then took the geometric mean, γ * prior_average + (1- γ) * new_observation, with $\gamma = 0.7$. The geometric mean is a method of summarizing sequential data, but provides lesser weight to older observations, as prior observations are decayed by γ at each time step.
- How many times the student reached a bottom out hint on this skill.
- How often the student was rapidly guessing, computed across all skills, defined as submitting responses less than 2 seconds apart on successive items. We took the geometric mean in the same manner as for response time.
- How often the student gave a rapid response, computed across all skills, defined as responding in a time frame that suggests a reading rate of over 400 words per minute. We took the geometric mean of this feature.
- How often the student reached a bottom out hint on 3 consecutive problems, computed across all skills; a 1 indicates the student requested the answer on 3 consecutive problems. We took the geometric mean of this feature.
- The name of the current skill.

We fit this model using logistic regression in SPSS. The first six terms were covariates, and final term was entered as a fixed effect (i.e., one parameter per skill). Note that we were unable to have user identity as a factor in this model, as that exceeded SPSS's capabilities; therefore statistical reliability would be somewhat overstated due to non-independence of student trials [6], and we therefore do not report

statistical reliability. Table 1 summarizes the model's accuracy. Each row denotes how well the model is doing after seeing the student solve a given number of problems on the skill. The second column indicates the percentage of the student-skill pairs that resulted in wheel spinning. When students first began a skill, 9.8% of the data included wheel spinning. For students who had not mastered a problem by the fifth attempt, 38.5% of the data indicated wheel spinning. The third column is R^2, a metric of model fit, which ranges from 0 (unable to predict the data) to 1 (perfect accuracy). Note that even before the student begins solving a problem on the skill, the model is able to account for 13% of the variation in wheel-spinning. The model is able to make use of the last four features listed above, the student's gaming behavior on other problems, and the identity of the current skill, since those do not depend on the student's performance on the current skill.

Table 1. Model performance for predicting wheel-spinning

# problems on this skill	Wheel-spinning %	Nagelkerke R^2	False positive%	False negative%
0	9.8%	0.13	0.3%	98.2%
1	9.8%	0.28	1.0%	88.6%
2	9.8%	0.39	1.6%	73.9%
3	20.5%	0.37	4.7%	60.5%
4	28.5%	0.41	9.2%	47.3%
5	38.5%	0.45	15.3%	33.7%
6	53.2%	0.44	27.0%	21.2%
7	67.5%	0.65	28.2%	10.2%
8	83.5%	0.85	3.9%	4.6%

The fourth column denotes false positives, that is, cases where the model predicts the student will wheel-spin, but in fact the student masters the skill within the first 10 practice opportunities. The model has a fairly low false percentage rate, mostly due to the imbalanced classes as, initially, wheel-spinning is a small minority of the data. The false positive rate continues to rise as wheel-spinning students constitute a larger and larger percentage of the dataset. However, in general, if the model asserts a student will wheel-spin, it is usually correct.

The fifth column denotes false negatives, the case where the model predicts the student will master the skill, but instead he wheel-spins. Initially, this rate is extremely high, mostly due to the model being unwilling to predict wheel-spinning, the minority class, on the basis of little data about the student's knowledge of this skill. The model does not do a good job at catching most of the cases when the student will wheel-spin, and is a bit conservative in its predictions. Thus, as an early warning system for preventing students from having frustrating problem-solving sequences without mastery, the detector still needs additional work. However, after students have solved 2 problems on a skill, it does a relatively good job at detecting wheel-spinning, and is potentially able to save students some frustration.

One question is what is model's source of power? Plotting the β values estimated by the logistic regression, the impact of the features is relatively stable across the models. The importance of the number of consecutive correct responses increases as the number of problems seen increases. This result makes intuitive sense; for example, a student with 0 correct in a row on the second problem is not in as much difficulty as a student with 0 correct in a row on the sixth problem, as the latter student has few remaining opportunities to get 3 right in a row. The student's normalized response time becomes relatively less important the more problems that are seen. Initially students who take relatively more time than average to complete a problem have a lower chance of wheel-spinning. This relationship is a bit surprising, but could perhaps be due to fast responses being ambiguous and indicating either the student is very skillful (and is likely to master the skill), or the student is just entering a random response (and is likely to wheel-spin). In general, the features related to performance on other skills become markedly less important the more problems the student solved on this skill. Again, this result makes intuitive sense as the fewer data available about this skill, the more useful data about the student's performance on other skills will be.

Beyond predicting wheel-spinning, we also explored its relationship with the negative behavior of gaming. We found that the 2491 students who never exhibited wheel-spinning had a mean gaming score of 0.013; the 366 students who always exhibited wheel-spinning had a mean gaming score of 0.163. Thus, students who wheel-spin are also likely to game. But does this relationship hold within a particular student; that is, when a student wheel-spins is he more likely to game than when he masters a skill? For the 1207 students who sometimes exhibited wheel-spinning, their mean gaming score was 0.104 when wheel spinning vs. only 0.017 when they mastered skills in a timely manner. These numbers are similar to the corresponding gaming values for students who always wheel-spun or always mastered quickly. This result strongly suggests that gaming and wheel-spinning are related. However, it leaves unresolved the direction of causality.

4 Contributions

The primary contribution of this paper is to identify a new problem in student modeling that is actionable by the tutorial decision-making module of an ITS. Most efforts in student modeling (e.g. [5, 7, 8]) and the 2010 KDD Cup on Educational Data Mining, focus on predicting student behavior at the level of individual responses. Although this approach clearly validates the student model, a reasonable question is why this problem is an interesting one in the first place, particularly from the standpoint of building an effective, adaptive ITS. Imagine our models had half of the error rate of the current state of the art, would that appreciably improve the performance of computer tutors? It is unclear what tutorial decisions would be affected by such better models, beyond slight refinements in the mastery learning model for when to consider the student done with a skill. In contrast, a strong model of wheel-spinning has clear implications for how to adapt instruction to the student. Consider Fig. 2 as one possible model for an ITS to incorporate a wheel-spinning detector, by modifying the typical ITS mastery learning cycle to not always present another problem in the event

of a student mistake. If the student is likely to wheel-spin, there is little point in providing another problem to the student as he is unlikely to master the skill. As problem solving is, statistically, a futile exercise for this learner at this time, doing something other than problem solving seems warranted. There are a variety of possible methods of instruction, including intervention by the teacher, peer tutoring, or incorporating stronger instruction into the ITS itself (e.g. [9]). We do not have sufficient data to prescribe a particular solution to wheel-spinning, but are willing to conclude that more problem solving is not a viable approach. In addition, wheel-spinning can be computed from log files and does not require human coders to train a model, and it also accounts for a moderate percentage (10% to 35%) of behavior.

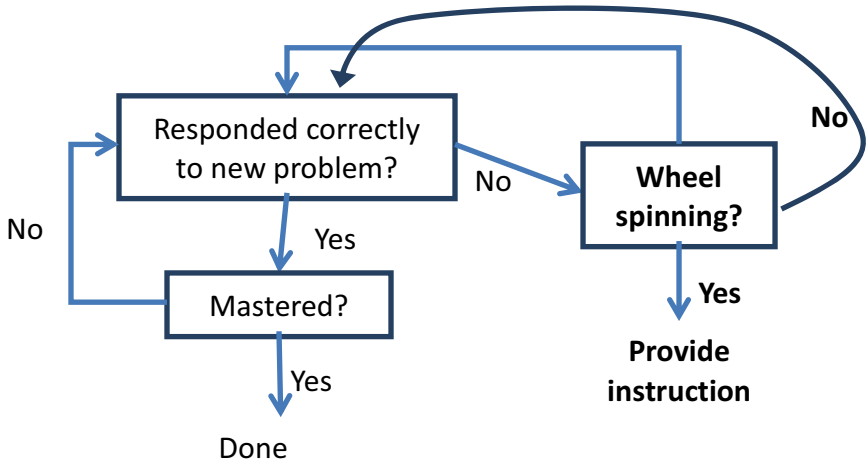

Fig. 2. Possible process model for incorporating wheel-spinning into an ITS

Finally, this paper provides a new approach for evaluating the impact of an intelligent tutoring system using Fig. 1 as a visualization. We can separate students into three categories. First, there are the 35% of the students in both CAT and ASSISTments who mastered the material with no mistakes; as they knew the skill before starting they did not directly benefit from the tutoring components. Second, there are the 9.8% (lower bound) to 35% (upper bound) of students who wheel-spun, they did not benefit from the tutor. The third group of students are those who *potentially* benefitted from the tutor, a group comprising 30% to 55% of the student population. We cannot determine whether these actually students benefitted or not, as perhaps they would have mastered the skill with simple pencil and paper practice. However, the upper bound on the percentage of students who could have been helped by the existing tutor is surprisingly low.

5 Conclusions and Future Work

This work is an initial attempt to model and understand wheel-spinning, and there are several unanswered questions. First, what does wheel-spinning look like in other

systems? We have examined two mathematics tutors and found broadly similar patterns of behavior. Does this problem exist in other tutorial domains? A second question is what is the proper unit of measure for the x-axis? This work used the number of problems, but perhaps total time spent would be a better indicator?

The second dimension of future work involves understanding the nature of wheel-spinning, and how generalizable the detector is. We constructed a set of predictive features based upon our beliefs as to what would be predictive, but there are hopefully additional predictors that can be brought to bear on this problem. A related issue is whether the predictors are consistent across tutoring systems; how well would a detector for ASSISTments work on CAT, and vice versa? Would such a detector generalize to a non-mathematics domain? One likely important step in this process is the proper normalization of the data, similar to what was done for response time.

The third area of work involves exploring the relationship between wheel-spinning and negative behaviors such as gaming [10] and off task behavior [11]. We found that gaming and wheel-spinning were correlated behaviors, but it is a question as to the direction of causality, as there are three plausible models:

- Gaming causes wheel-spinning. Students are not taking problems seriously and requesting hints they perhaps do not need. As a result of help requests being scored as incorrect responses, students wheel-spin and do not achieve mastery.
- Wheel-spinning causes gaming. Students who do not understand the material are unable to solve the problems and become frustrated. Such students have no way to proceed other than requesting many hints, since many ITS do not have strong instructional components.
- Gaming and wheel-spinning are symptoms that are affected by a common cause.

These models have very different implications, as the first model suggests trying to affect the student's mood directly is a viable approach. The second model suggests that instruction is more likely to be beneficial. In reality, there is probably a mixture of both behaviors going on. It is interesting to note that the first work on remediating gaming had a positive effect [10], but included components that both attempted to discourage gaming, but also added instructional support beyond what was previously available in the tutor. Controlled studies that provide instruction to students who are likely to wheel-spin would be useful for disambiguating which of the two candidate hypotheses is closer to reality.

We see two clear consequences of this work. First, it is perhaps not wise to use all data for model-fitting purposes when training a student model. Since most student models assume a fixed probability of learning a skill, long sequences of problems by wheel-spinning students are likely to underestimate the learning rate for the average student. The distribution of learning rates can be thought of as bimodal with a group of wheel-spinning student-skill pairs clustered near 0. Thus, work on detecting contextual factors that affect learning [e.g., 12] is a welcome development.

The second consequence is that ITS designers should develop some fallback for failures of mastery learning. The simplified mastery learning cycle of "present problems until mastery" does not work for many learners, even with the assistance available in two popular tutors. Some modification or automated intervention is warranted if we wish to avoid frustrating learners.

Acknowledgements. This work was supported by the National Science Foundation (grant DRL-1109483) to Worcester Polytechnic Institute. The opinions expressed are those of the authors and do not necessarily represent the views of the Foundation.

References

1. Koedinger, K.R., et al.: Intelligent Tutoring Goes To School in the Big City. International Journal of Artificial Intelligence in Education 8, 30–43 (1997)
2. Bloom, B.S.: Human characteristics and school learning. McGraw-Hill (1976)
3. Frick, T.W.: A comparison of three decision models for adapting the length of computer-based mastery tests. Journal of Educational Computing Research 6(4), 479–513 (1990)
4. Corbett, A., Anderson, J.R.: Knowledge tracing: Modeling the acquisition of procedural knowledge. User Modeling and User-Adapted Interaction 4, 253–278 (1995)
5. Beck, J.E., Chang, K.-M.: Identifiability: A Fundamental Problem of Student Modeling. In: International Conference on User Modeling, Corfu, Greece (2007)
6. Menard, S.: Applied Logistic Regression Analysis. Quantitative Applications in the Social Sciences. Sage Publications (2001)
7. Pardos, Z., et al.: Analyzing fine-grained skill models using bayesian and mixed effect methods. In: Thirteenth Conference on Artificial Intelligence in Education. IOS Press (2007)
8. Chi, M., et al.: Instructional Factors Analysis: A Cognitive Model For Multiple Instructional Interventions. In: Proceedings of Educational Data Mining (2011)
9. de Koning, K., et al.: Model-based reasoning about learner behaviour. Artificial Intelligence 117, 173–229 (2000)
10. Baker, R.S.J.d., et al.: Adapting to When Students Game an Intelligent Tutoring System. In: Ikeda, M., Ashley, K.D., Chan, T.-W. (eds.) ITS 2006. LNCS, vol. 4053, pp. 392–401. Springer, Heidelberg (2006)
11. Baker, R.S.J.d.: Modeling and Understanding Students' Off-Task Behavior in Intelligent Tutoring Systems. In: Proceedings of ACM CHI 2007: Computer-Human Interaction (2007)
12. Baker, R.S.J.d., Goldstein, A.B., Heffernan, N.T.: Detecting the Moment of Learning. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010, Part I. LNCS, vol. 6094, pp. 25–34. Springer, Heidelberg (2010)