Detecting When Students Game The System, Across Tutor Subjects and Classroom Cohorts

Ryan Shaun Baker, Albert T. Corbett, Kenneth R. Koedinger, Ido Roll

Human-Computer Interaction Institute, Carnegie Mellon University,
5000 Forbes Avenue, Pittsburgh, PA, 15217, USA
{rsbaker, corbett, koedinger, idoroll}@cmu.edu

Abstract. Building a generalizable detector of student behavior within intelligent tutoring systems presents two challenges: transferring between different cohorts of students (who may develop idiosyncratic strategies of use), and transferring between different tutor lessons (which may have considerable variation in their interfaces, making cognitively equivalent behaviors appear quite different within log files). In this paper, we present a machine-learned detector which identifies students who are "gaming the system", attempting to complete problems with minimal cognitive effort, and determine that the detector transfers successfully across student cohorts but less successfully across tutor lessons.

1 Introduction and Prior Work

In the last couple of decades, there has been considerable work in creating educational systems that adapt to their users – offering help and feedback targeted to a student's specific cognitive or motivational needs. However, just as educational systems can adapt to their users, users can adapt to their educational systems, sometimes in ways that lead to poorer learning [2,5]. For instance, students who game the system, attempting to perform well in an educational task by systematically exploiting properties and regularities in the system used to complete that task, rather than by thinking about the material, learn less than other students [2]. Examples of gaming include systematic guessing, and repeatedly requesting help until the system gives the answer. It may be possible to substantially improve learning environments' educational effectiveness by adapting to how students choose to use the learning environment. In order to do this, we need to be able to detect when a student is selecting strategies that lead to poorer learning.

In [1], we presented a Latent-Response Model [4] that accurately detected if a student was gaming the system, within a specific tutor lesson, cross-validated across students in 4 classes. This model distinguished "GAMED-HURT" students who gamed the system in a fashion associated with poor learning both from students who were never observed gaming, and from "GAMED-NOT-HURT" students who gamed in a different fashion not associated with poor learning. The model did so by first predicting whether each individual student action was an instance of gaming (using tutor log files), and then aggregated these predictions to predict what proportion of

time each student was gaming (comparing the predicted proportions to data from classroom observations). The classifier's ability to distinguish gaming was assessed with A' values, which give the probability that if the model is given one gaming student and one non-gaming student, it will accurately identify which is which [3].

A model in this framework consists of features selected from linear, quadratic, and interaction effects on a set of 26 base features describing a student action (for instance, what interface widget it involved and how long it took), and its historical context (for instance, how many errors this student made on this skill in past problems). The model presented here improves on the model reported in [1] in three fashions: First, by adding two features to the set used in [1], in order to represent asymptotic skills (which students on the whole either knew before starting the tutor, or failed to learn while using the tutor). Second, by switching from using forward selection to select model features to testing a set of search paths constrained by fast correlation-based filtering [6] (in both cases, Leave One Out Cross Validation was used to prevent over-fitting). Third, by switching from treating both types of gaming as identical during training to training to detect just GAMED-HURT students, considerably improving our model's ability to distinguish between types of gaming, $\Delta Z=6.57$, $p<0.01$. After these changes, our model was significantly better than chance at distinguishing GAMED-HURT students from non-gaming students (within the original classroom cohort and lesson), $A' =0.85$, $p<0.01$, and at distinguishing GAMED-HURT students from GAMED-NOT-HURT students, $A' =0.96$, $p<0.01$.

Though this detector is effective within a single population and tutor lesson, it will be more useful if it can generalize across student populations and cognitive tutor lessons (or even across types of interactive learning environments). There appear to be multiple ways to game a given system, and we have observed students teaching each other new strategies for gaming – therefore, different cohorts of students may game differently. Similarly, different tutor lessons often have different patterns of interaction, because of differences in subject matter. In this paper, we present work towards detecting gaming in a fashion robust to differences between tutor lessons and classroom cohorts, through analyzing how well a model trained on one population or lesson transfers to other populations and lessons, and how the features that correlate to gaming differ across data sets.

2 Detecting Gaming Across Classroom Cohorts

In this section, we discuss how well our detector transfers between our original student cohort (termed the 2003 cohort) and a newly recruited cohort of students (termed the 2004 cohort). At a surface level, the two cohorts were similar: both were drawn from students in 8th and 9th grade non-gifted/non special-needs cognitive tutor classrooms in the same middle schools in the suburban Pittsburgh area. However, our observations suggested that the two cohorts behaved differently. The 2004 cohort gamed 88% more frequently than the 2003 cohort, $t(175)=2.34$, $p=0.02$, but a lower proportion of the gaming students had poor learning, $\chi^2(1, N=64)=6.01$, $p=0.01$. This data does not directly tell us whether gaming was different in kind between the two

Table 1. Our model's ability to transfer between student cohorts. Boldface signifies both that a model is statistically significantly better within training cohort than within transfer cohort, and that the model is significantly better than the model trained on both cohorts.[1]

Training Cohort	G-H vs no game, 2003 cohort	G-H vs no game, 2004 cohort	G-H vs G-N-H, 2003 cohort	G-H vs G-N-H, 2004 cohort
2003	*0.85*	*0.76*	***0.96***	0.69*
2004	*0.77*	***0.92***	*0.75*	***0.94***
Both	*0.8*	*0.86*	*0.85*	*0.85*

populations – however, if gaming differs substantially in kind between populations, two populations as different as these are likely to manifest such differences, and thus these populations provide us with an opportunity to test whether our gaming detector is robust to differences between distinct cohorts of students.

The most direct way to evaluate transfer across populations is to see how successfully the best-fit model for each cohort of students fits to the other cohort. As shown in Table 1, a model trained on either cohort could be transferred as-is to the other cohort, without any re-fitting, and perform significantly better than chance at detecting GAMED-HURT students (marginally significantly better at distinguishing them from GAMED-NOT-HURT students in the 2004 cohort; significantly better in all other comparisons). However, in 3 of the 4 comparisons, the models were statistically significantly better in the student population within which they were trained than when they were transferred to the other population of students.

It was also possible to train a model, using the data from both student cohorts, which achieved a good fit to both data sets, shown in Table 1. This model was significantly better than chance in all 4 comparisons conducted. However, models trained in single cohorts did better than the unified model, in 3 of the 4 comparisons.

3 Detecting Gaming Across Tutor Lessons

In this section, we discuss how well our detector transfers between two tutor lessons, within a single student population. One lesson (the "scatterplot" lesson) involved creating and interpreting scatterplots of data; the other lesson (the "geometry" lesson) involved computing the surface area of 3D solids. Both lessons were drawn from the same middle-school mathematics curriculum and were designed using the same general pedagogical principles, although the scatterplot lesson had a greater variety of widgets and a more linear solution path. Our observers did not notice substantial differences between the types of gaming they observed in these two lessons. Overall, the same students gamed between lessons -- a student's frequency of gaming was also correlated across lessons, $r=0.22$, $p=0.02$.

The most direct way to evaluate transfer across lessons is to see how successfully

[1] All numbers are A' values. Italics denote a model which is statistically significantly better than chance ($p<0.05$); asterisks (*) denote marginal significance ($p<0.10$).

Table 2. Models trained on the scatterplot lesson, the geometry lesson, and both lessons together. All models trained using only the 2004 students.[1] Boldface denotes the model(s) which are statistically significantly best in a given category.

Training Lesson	G-H vs no game, SCATTERPLOT	G-H vs no game, GEOMETRY	G-H vs G-N-H, SCATTERPLOT	G-H vs G-N-H, GEOMETRY
SCATTERPLOT	***0.92***	0.55	***0.94***	0.63
GEOMETRY	0.53	***0.80***	0.41	***0.90***
BOTH	***0.82***	***0.77***	0.70*	*0.82*

the best-fit model for each tutor lesson fits to the other tutor lesson. As shown in Table 2, the results were poor. Though both models were significantly better than chance within the training lesson, neither model was significantly better than chance when transferred to the other lesson. It was possible to train a model, using both data sets, which achieved a good fit to both data sets, as shown in Table 2. This model was significantly better than chance on 3 of 4 measures (and was marginally significant on the fourth); however, on 2 of 4 measures it was statistically significantly worse than a model trained on one lesson alone. But while this unified model performed well in the units it was trained in, it transferred very poorly to the 2003 cohort of students using the scatterplot tutor, only reaching A'=0.54,p=0.77 (G-H versus non-gaming) and A'=0.54,p=0.78 (G-H versus G-N-H). This result is surprising, considering that a model trained just on the 2004 cohort using the scatterplot tutor was quite effective at detecting gaming within the 2003 cohort (see Table 1). Hence, although we can develop a unified model at this point, our modeling approach has not yet delivered a unified model which transfers across lessons in a generalizable fashion.

But why not? The difference in gaming between these lessons is small enough that our observers did not notice a qualitative difference in gaming between them. Additionally, the top candidate features considered for each lesson (which are highly correlated to gaming but not to each other) appear conceptually similar (see Table 3). In both sets, gaming corresponds to errors and repeated quick actions. However, the top 6 features for scatterplots averaged an unimpressive correlation of 0.06 to gaming in the geometry data set, and the top 6 features for geometry averaged a correlation of 0.09 to gaming in the scatterplot data set, suggesting that the difficulty in transferring between models is not just an artifact of the specific features chosen during model selection. It is possible that the overall strategic choice underlying gaming is consistent across the two lessons, but that the interface and pedagogical differences between the two lessons may be causing our models to differ considerably at the detailed grain size our approach relies upon to make predictions.

Table 3. Top 3 non-intercorrelated GAMED-HURT features in each lesson (2004 data).

SCATTERPLOT
1) Several quick actions in a row
2) A high percentage of errors on skills that involve popup menus (ie multiple choice)
3) Quick actions on problem steps that need a numerical answer

GEOMETRY
1) Requesting help several actions in a row on skills the student has a history of getting wrong
2) Several very brief help requests in quick succession
3) Several very quick errors in succession

4 Discussion and Conclusions

In this paper, we have presented a system that detects when a student is gaming the system. This system transfers successfully across cohorts of students. However, the same detector can not, at this point, transfer without re-training to different tutor lessons. Furthermore, training data from two lessons together does not produce a model which can transfer across student cohorts. Despite this, detectors for different lessons are detecting qualitatively similar behavior. One approach would be use our knowledge of what actions are gaming in different lessons to develop a system that maps from a tutor interface to gaming actions. However, given that our approach can train successful models for fairly different tutor lessons, it may not actually be necessary to make individual models that can generalize across lessons. For example, if the detector is deployed in a year-long curriculum, it may be possible to develop interventions which guide students to stop gaming, where the effect s maintained even after the intervention is no longer present. In this event, we would only need to detect gaming in a few lessons during the course of a curriculum, and could train a detector for each of those lessons. This approach would not afford rapidly extending our detector to new curricula, but may still be quite effective in improving student learning. Regardless, a gaming detector such as ours will only be useful if combined with an intervention that persuades students to change how they use the tutor. If the tutor responds to gaming in a fashion that gives students an incentive to learn how to game the gaming detector, the gaming detector will quickly become ineffective. Systems that detect intentional mis-use must adapt in a fashion that makes it in the student's interest to use the software appropriately.

Acknowledgements. We would like to thank James Fogarty, Vincent Aleven, Angela Wagner, Tom Mitchell, Brian Junker, Amy Hurst, Cristen Torrey, and Amy Ogan for helpful suggestions and assistance. This work was funded by an NDSEG Fellowship.

References

1. Baker, R.S., Corbett, A.T., Koedinger, K.R. Detecting Student Misuse of Intelligent Tutoring Systems. Proceedings of the 7th International Conference on Intelligent Tutoring Systems (2004), 531-540.
2. Baker, R.S., Corbett, A.T., Koedinger, K.R., Wagner, A.Z. Off-Task Behavior in the Cognitive Tutor Classroom: When Students "Game the System". Proceedings of ACM CHI 2004: Computer-Human Interaction (2004) 383-390
3. Donaldson, W. Accuracy of d' and A' as Estimates of Sensitivity. Bulletin of the Psychonomic Society Vol. 31(4) (1993) 271-274.
4. Maris, E. Psychometric Latent Response Models. Psychometrika vol.60(4) (1995) 523-547.
5. Stevens, R., Soller, A., Cooper, M, Sprang, M. Modeling the Development of Problem-Solving Skills in Chemistry with a Web-Based Tutor. Proceedings of the 7th International Conference on Intelligent Tutoring Systems (ITS 2004), 580-591.
6. Yu, L., Liu, H. Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution. Proc. of the Intl. Conference on Machine Learning (ICML-03), 856-863.