A classifier to detect student 'gaming' of a medical education system

Olga Medvedeva[1], Adriana M.J.B. de Carvalho[2], Ryan S.J.d. Baker[3], Rebecca S. Crowley[1]

[1] Department of Biomedical Informatics, University of Pittsburgh School of Medicine, Pittsburgh PA
[2] Human Computer Interaction Institute, Carnegie Mellon University
[3] Teachers College, Columbia University, New York, NY

Abstract

We report on the training, testing and application of a classifier for detecting 'gaming the system' – a well documented behavior in which students exploit the properties of an educational system to succeed without learning. Using outcomes based on expert judgments of user-system interaction sequences, we trained a decision tree classifier, using ten-fold cross validation. The area under the ROC curve was 0.72, significantly better than chance: Z=10.90, two-tailed p<0.001. We applied the resulting gaming detector model to 160 hours of interaction data from 70 users in four different studies. Results show that gaming behaviors are frequently observed, and that gaming frequency is negatively correlated with learning gains. The resulting gaming detector will be used in the SlideTutor system to automate gaming detection, and intervene when gaming is observed.

Introduction

Lack of effort is an important negative factor when students learn using a computer based system, and is associated with disengaged behaviors. One specific type of disengaged behavior called 'gaming the system' has been the subject of increasing attention recently. Gaming the system is defined as the attempt "to succeed in an educational environment by exploiting properties of the system rather than by learning the material" [3]. An important goal for educational systems is to automatically identify such students in order to intervene and alter their behavior.

Recent research has shown that gaming can be automatically detected by using machine learning methods to train classifiers of student-system interaction data [2,3]. Results of this research has shown that gaming (1) can be detected in a range of educational systems [3,5], (2) is associated with decreased learning gains [1], and (3) is amenable to intervention [4]. Classifiers have been built using different methods for labeling training data, including field observation and text replay. The text replay method has been shown to produce gaming detectors with high performance in a fraction of the time of other methods [5]. We used this methodology to develop the first gaming classifier in a medical education system.

System Description

SlideTutor is an intelligent tutoring system that teaches surgical pathology [6]. Several previous evaluation studies of our system have demonstrated its efficacy in improving diagnostic accuracy [7], reporting completeness [8,9] and correctness [8], and metacognitive performance [10]. The system trains students to accurately perform two related tasks – diagnosis and reporting. In the diagnostic component (Figure 1A), students inspect pathology slides using a virtual microscope, change magnification, mark the image to identify visual features, specify qualities of these features, create hypotheses and ultimately make one or more diagnoses. In the reporting component, (Figure 1B) students write diagnostic reports on slides, identifying prognostic features, and perform specific actions such as determining if a margin is involved by cancer, or measuring lesion depth.

Each problem or case is composed of specific intermediate steps (subgoals or skills) of specific subgoal types. For example, one intermediate step in solving a case would be to identify the feature (subgoal type) of 'subepiedermal blister' (subgoal). The system evaluates the student action at each subgoal and determines whether it is CORRECT or INCORRECT by comparing it with its expert module. Correct actions generate no system intervention. Incorrect actions are identified immediately (often by coloring them red or flashing) and the system provides an explanation and additional corrective information. When lost, students can request a next-best-step (HINT) which is context dependent, and specific to the state of the problem and the student's skill level. All student actions and system responses are time-stamped and stored in an Oracle database. During the past ten years of research, we have created many versions of the SlideTutor system to address various educational and research goals. All of these systems share a specific framework for user interaction (based on the 'cognitive tutor' design) which we took advantage of to create a generalized gaming model.

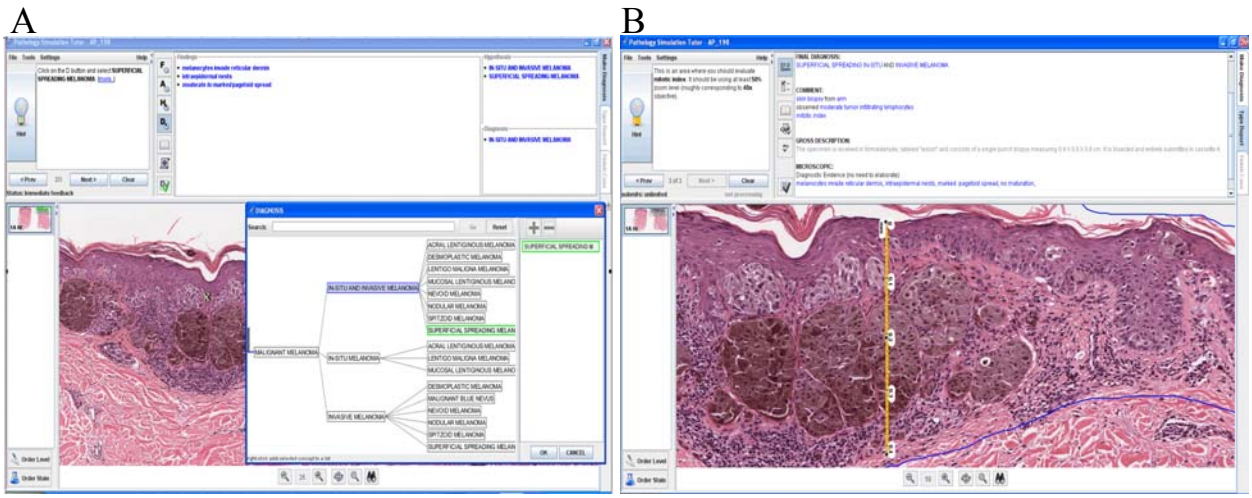

Figure 1. (A) Diagnostic Tutoring Component. Students examine virtual slides (bottom left), create diagnostic reasoning diagram (top right) by combination of marking image and selecting from interactive feature and diagnosis trees (bottom right) and may request hints (top left) when lost. **(B) Report tutoring component.** Students perform various inspection tasks such as measurement (bottom), and write reports in free text (top right) They may also request hints (top left) when lost.

Methods

Modifications to existing gaming detection model

We followed the approach described by for developing gaming detectors for mathematics tutoring systems. In summary, we reused 15 of 26 parameters described in previous research [2,3].

Features from previous research:

- Tutor Response (nominal: correct incorrect, hint)
- Was tutor response incorrect (binary)
- Was tutor response correct? (binary)
- Was tutor response a hint? (binary)
- Time taken for previous 3 actions, normalized for all users
- Time taken for previous 5 actions, normalized for all users
- Number of errors made on this subgoal across all problems
- Average number of hint requests made on this subgoal across all problems
- Average number of errors made on this subgoal across all problems
- Total time spent on this subgoal across all problems
- Number of previous 3 actions that were on same subgoal
- Number of previous 5 actions that were on same subgoal
- Number of the previous 8 actions that were help requests
- Number of the previous 5 actions that were errors
- Has student made at least 3 errors on this subgoal in this problem (binary)

New Features:

- Total time spent on this subgoal type across all problems, normalized across all users
- Are the last 2 actions HINT and INCORRECT in sequence? (binary)
- Are the last 3 actions INCORRECT, HINT and CORRECT in sequence? (binary)

Table 1. Model Features

From the set of features used in other ITS gaming detectors, we selected features common to all of our tutoring systems, and excluded any nominal features related to specific skills (for example skill name, lesson, etc), because of the large scope of our domain. Additionally we included other interface actions that suggest gaming the system but are unique to our visual classification tutoring systems. These include (1) protracted digital slide exploration, (2) guessing of features and their locations, (3) indiscriminate browsing of feature and diagnosis trees, (4) playing with tools such as the measurement tool and (5) ignoring the hint suggestions. Finally, we elected to more carefully separate users who appear to be using hints to walk through an expert solution [11] from those who are using hints in a less systematic manner, because the former group appears to be learning by example.

Datasets

Data for model training and testing (Table 2) was extracted from an Oracle database which stores all user-system communication for all of our research studies [12]. Models were trained on data collected for a study of resident physicians using the diagnostic tutoring component for cases of inflammatory diseases of skin [10]. We then applied the model to a larger dataset which also included three other studies: *Study 2*- a long term study of practicing community pathologists using diagnostic ITS with cases of melanocytic nevi and melanomas [9], *Study 3*- a short term study of resident physicians using diagnostic ITS with cases of subepidermal blistering diseases [7], and *Study 4* - a short term study of resident physicians using reporting ITS with cases of

melanocytic nevi and melanomas [8]. In all cases, we only included sessions in which the tutor was used in the immediate feedback condition. We also excluded the first case for each student because it typically takes one case to learn the interface.

Labeling of data for training set

We reproduced the standard method for training gaming detectors [5], using the same expert human judge (the second author) who has labeled data for gaming detectors in mathematics ITS [5]. From data extracted from study #1 (Table 2), we randomly selected 294 sequences of 5 student actions at subgoals which generate tutor responses, for a total of 1303 records which were manually labeled by the human judge. To capture human judgments, we slightly modified an existing system for Text Reply [5] (Figure 2). We added a "details" link to each action to provide additional information on what the user was doing between subgoal actions. For example, on the fourth action shown in Figure 3, before entering 'sclerosis' as evidence, the user (1) opened and closed the error message window, (2) closed the finding tree window, (3) moved the application window, (4) further explored the slide, (5) opened, expanded, and browsed the Findings tree and (6) selected sclerosis from it. This set of actions took 60.5 seconds. This detailed information helps the human judge to determine whether the user was gaming the system, prior to the student action.

Model training

The resulting 18 features were used to train the J48 decision tree classifier in Weka 3.6 [14]. We used ten-fold cross validation, and computed ROC curves to determine model performance, and the Z-test to determine whether classification performance is better than chance.

Analysis of gaming across studies

The resulting gaming detector was applied to all other datasets to characterize the frequency of gaming. For each student, we correlated frequency of gaming with the learning gains obtained from pre-test to post-test comparisons. We then separated students who gamed above the median (termed 'gamers') from those who gamed below the median ('not-gamers'), and separated the gamers into two groups by median learning gain (termed 'gamed-hurt' and 'gamed-not hurt') All analyses were performed in Matlab.

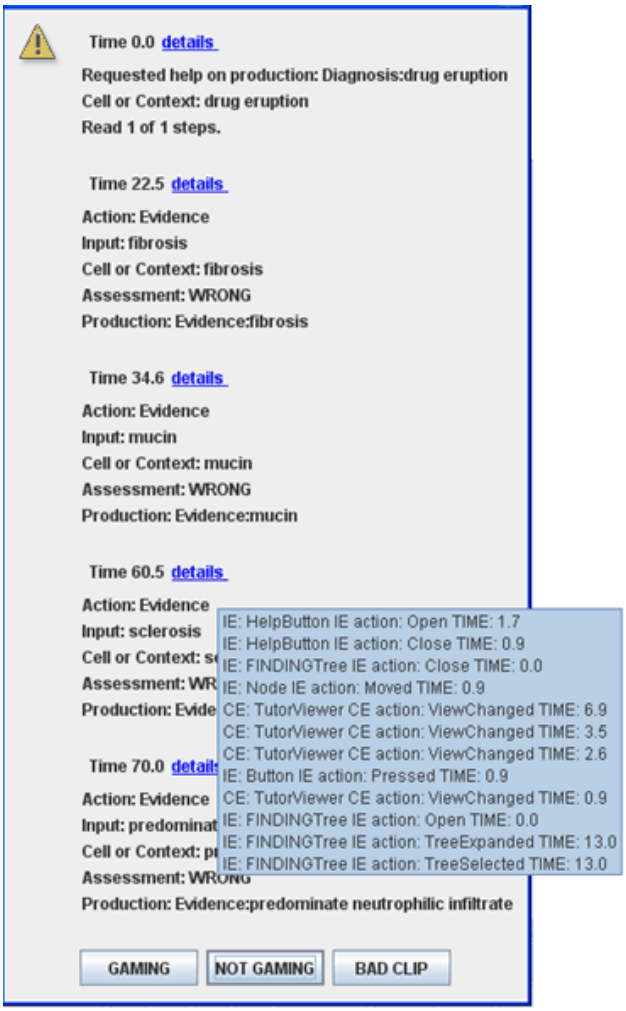

Figure 2. Text replay labeling by human judge

Study	Used for Training	# Users	# Records	Mean Records per User (SD)	Median Learning Gain/User	Median Gaming/ User	# (%) Gamed/ Hurt	# (%) Gamed/Not Hurt
1	Yes	23	10269	447 (112)	18%	6%	8 (35%)	4 (17%)
2	No	6	2452	510 (232)	75%	2%	2 (33%)	1 (17%)
3	No	21	14329	682 (188)	48%	5%	7 (33%)	4 (19%)
4	No	20	5091	255 (105)	61%	1%	3 (15%)	7 (35%)

Table 2. Summary of Datasets with Model Application Results

Results

Model Performance

First, we calculated the J48 classifier performance on the labeled training set. The area under the ROC curve (AUC) for 10-fold cross-validation was 0.72 and is significantly better than chance: Z=10.90, two-tailed p<0.001. The performance is sufficient for use in "fail-soft" interventions that are non-harmful when occasionally mis-assigned.

Analysis of datasets for gaming

We then applied the gaming detector model to data from all four studies to investigate the frequency of gaming and the relationship of gaming behaviors to learning gains. The transferability of gaming detectors between units in a specific intelligent tutoring system has previously been established [3].

Table 2 shows the number of users, records and mean number of records per user in each study, along with the median learning gain/user, and median gaming frequency/user. A total of 70 users, 160 hours and more than 30K records were analyzed.

The overall frequency of gaming in our tutoring systems varied from 1-6% across all four studies, which appears to be quite similar to the frequency of gaming reported in mathematics tutoring systems used by middle-school and high-school children, determined by both model application [2,3], text replays[5], and human judgments during field observation [1].

Relationship of Gaming to Learning

Figure 3 shows the relationship of user's learning gain with gaming frequency during the tutoring sessions for data from all of the studies. The Pearson Correlation Coefficient was -0.26, $t_{(1,138)}$=3.16, p=0.028. The negative correlation indicates that the greater the frequency of gaming, the less the user learned.

Not all users who game the system experience a negative outcome. We classified gamers by learning gains based on a median threshold, and found that in the majority of our studies – those identified as gamers were 'hurt' by their gaming (e.g. those with less than median learning gains), more commonly than 'not hurt' (Table 2). Distinguishing between these two groups is important, because we want to intervene among students who are gaming and not learning, and therefore would tolerate some

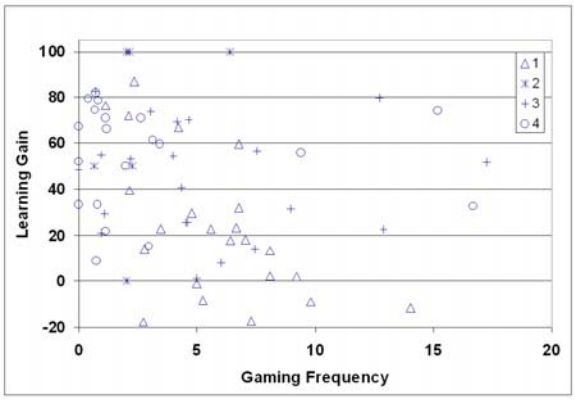

Figure 3. Correlation of learning gain (pre-test to post-test) with gaming frequency. Each point represents one student from study 1(Δ), study 2 (X), study 3 (+) or study 4(○).

misclassification of those who the model classifies as gaming, but are nonetheless learning in the system.

The learning gains we use are outcomes measures which we do not have access to until after the tutoring sessions are completed. Therefore, we investigated whether our model can predict learning outcome based on gaming frequency.

Using the gaming frequency per user from all studies as a model predictor and the binarized learning gains, where the users learning gain is 1 if he/she gained more than median, as an outcome, we calculated the AUC for all the studies. When we include all users, the ROC AUC is 0.69 (SE=0.06), Z=3.02 with p=0.001. However, if we exclude from consideration users in the 'gamed-not hurt' category, the AUC is 0.85 (SE=0.05), Z=7.23 with p<0.001, suggesting that we are better able to predict learning level among those who game and are negatively impacted, the group that we specifically wish to address.

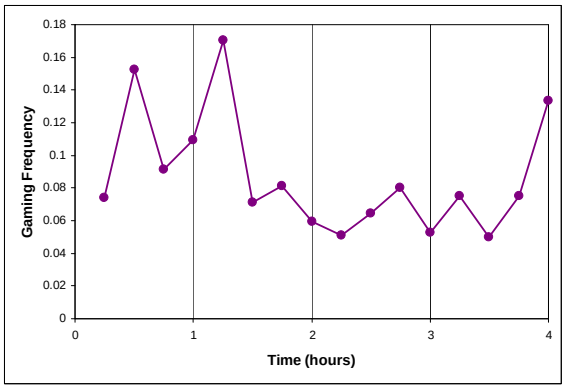

Figure 4. Gaming frequency over time

Next, we analyzed the frequency of gaming over time, and found that our model suggests that gaming is more frequent early in the educational sessions (Figure 4), which differs from what has been reported in other ITS [15]. We also separated users by learning gains, but found no relationship between the temporal patterns and learning outcomes.

Discussion

With this study, we demonstrate the potential for automatically identifying an important disengaged behavior, gaming the system, among users of SlideTutor. To our knowledge, this is the first attempt to automatically identify this behavior among users of a medical education system.

The frequency of gaming detected by the classifier in our ITS is comparable to what has been reported in other tutoring systems used by middle-school and high-school children [2,3,5]. This result is in some ways surprising, since the predicted high skill and motivation level of our users (physicians) would seem at odds with this finding. In other ways, this result is not surprising to us at all. Anecdotally, we often observe occasional users who seem to exert minimal effort and end up learning very little from the system. In the past, we have found it difficult to identify these students up front [13]. The gaming detector we developed in this paper will now allow us to detect gaming as students use the system, enabling us to intervene in these cases. In a follow-up study, we will test the effect of intervention on learning gains.

This study also raises an important unanswered question: What separates students who game the system and learn poorly from those who seem not to be impacted? It seems likely that the actions we detect as 'gaming' have heterogenous causes. For example, frequent use of the hint button may suggest that students prefer to take the easiest way to get through the lesson, even if it means they do not exert the effort needed to 'learn by doing'. But repeatedly 'hinting through' a problem could also be an effective way to 'learn by observing' the expert solution. This might be more common early in the session until users feel confident in their abilities.

Acknowledgements

This work was supported by a grant from the National Library of Medicine (2 R01 LM 007891).

References

1. Baker RS, Corbett AT, Koedinger KR, Wagner AZ. Off-Task Behavior in the Cognitive Tutor Classroom: When Students "Game The System". Proc ACM CHI 2004: Computer-Human Interaction: 383-390.
2. Baker RS, Corbett AT, Koedinger KR. Detecting Student Misuse of Intelligent Tutoring Systems. Proc 7th Intl Conf on ITS, 2004; 531-540.
3. Baker, RSJd Corbett AT, Roll I, Koedinger KR. Developing a Generalizable Detector of When Students Game the System. User Model User-Adapt Inter, 2008; 18:287-314.
4. Baker RSJd, Corbett AT, Koedinger KR, Evenson E, Roll I, Wagner AZ, Naim M, Raspat J, Baker DJ, Beck J. Adapting to When Students Game an Intelligent Tutoring System. Proc 8th Intl Conf on ITS, 2006; 392-401.
5. Baker RSJd, de Carvalho AMJA. Labeling Student Behavior Faster and More Precisely with Text Replays. Proc 1st Intl Conf on Educ Data Mining, 2008; 38-47.
6. Crowley RS, Medvedeva O. An intelligent tutoring system for visual classification problem solving. Artificial Intelligence in Medicine. 2006; 36:85-117.
7. Crowley RS, Legowski E, Medvedeva OM, Tseytlin E, Roh E, Jukic D. Evaluation of an Intelligent Tutoring System in Pathology: Effects of External Representation on Performance Gains, Metacognition, and Acceptance. J Am Med Inform Assoc, 2007; 14:182-190.
8. Saadawi GM, Tseytin E, Legowski E, Jukic D, Castine M, and Crowley RS. A natural language intelligent tutoring system for training pathologists: implementation and evaluation. Adv Health Sci Educ, 2008; 13:709-722.
9. Crowley RS, Gryzbicki D, Legowski E, 1, Wagner L, Castine M, Medvedeva O, Tseytlin E, Jukic D, Raab S. Use of a Medical ITS Improves Reporting Performance among Community Pathologists. Accepted to Proc 10th Intl Conf on ITS 2010.
10. Saadawi GM, Azevedo A, Castine M, Payne V, Medvedeva O, Tseytlin E, Legowski E, Jukic D and Crowley RS. Factors Affecting Feeling-of-knowing in a Medical Intelligent Tutoring System – the Role of Immediate Feedback as a Metacognitive Scaffold. Adv Health Sci Educ, 2010; 15:9-30
11. Shih B, Koedinger KR, Scheines R. A response time model for bottom-out hints as worked examples. Proc 1st Intl Conf on Educ Data Mining, 2008; 117-126.
12. Medvedeva O, Chavan G, Crowley RS. A data collection framework for capturing ITS data based on an agent communication standard. Proc AAAI 2005; Technical Report WS-05-02, AAAI Press 2005; 23-30.
13. Yudelson MV, Medvedeva O, Legowski E, Castine M, Jukic D, and Crowley RS. Mining student learning data to develop high lever pedagogic strategy in a medical ITS (2006). Educ Data Mining Workshop Proc AAAI 2006. Tech Report WS-06-05. AAAI press, 2006; 82-90.
14. http://www.cs.waikato.ac.nz/ml/weka
15. Beck JE. Engagement tracing: using response times to model student disengagement. Proc 12th Intl Conf on AIED, 2005; 88-95.