How quickly can wheel spinning be detected?

Noboru Matsuda
Texas A&M University
4232 TAMU
College Station, TX 77843
Noboru.Matsuda@tamu.edu

Sanjay Chandrasekaran
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15213
sanjayc@andrew.cmu.edu

John Stamper
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15213
jstamper@cs.cmu.edu

ABSTRACT
We have developed a wheel spinning detector for cognitive tutors that uses a simplified method compared to existing wheel spinning detectors. The detector reads a sequence of the correctness of applying particular skill performed by a student using the cognitive tutor. The response sequence is first fed to Bayesian knowledge tracing to compute a sequence of probability of mastery at each time a skill was applied. The detector uses a neural-network model to make a binary classification for a response sequence into wheel-spinning and none-wheel spinning. To test the accuracy of the detector, we validated the detector using learning interaction data taken from a school study where students used a Geometry cognitive tutor. Human coders manually tagged the data to identify wheel spinning. The results show that the neural-network based detector has high recall (0.79) but relatively low precision (0.25) when combined with Bayesian knowledge tracing that detects mastery cases. The result suggests that the neural-network based detector is practical and has a potential for scalable use such as adaptive online course where cognitive tutors are embedded into online courseware.

Keywords
Wheel spinning; detector; neural network; Intelligent tutoring system; student modeling

1. INTRODUCTION
Cognitive tutors provide mastery learning on cognitive skills [3]. Mastery learning is controlled by a student-modeling technique called knowledge tracing [2] that computes the likelihood of mastering individual cognitive skills to be learned. The output from the knowledge tracer is used to compute an optimal sequence of training problems in such a way a student will achieve the mastery for all cognitive skills quickly [4].

One of the challenges under the paradigm of model-tracing based mastery learning happens when the student model does not detect a mastery within a reasonable amount of time. From the students' point of view, this means that they are continuously posed problems one after another for considerably long time. This phenomenon is called *wheel spinning* that has been coined by Beck and Gong [1].

Wheel spinning, by definition, means a situation in which a student does not reach to a pre-defined mastery level according to the mastery estimation computed by the knowledge-tracing algorithm. Although some students may eventually reach mastery only after working on a considerably many number of problems, it is not practical to assume that students would be persistent under such situation. When students do not see any improvement in their performance and the system merely provide more problems, then they would quickly get frustrated and lose their motivation. It is therefore quite important to detect wheel spinning as soon as possible. A reliable student-modeling technique to predict wheel spinning is there required.

The goal of current study is to develop a detector that detects a risk of wheel-spinning at an early phase of learning in the context of cognitive tutoring. The simplicity and scalability of the technology is one of the most important issues. We therefore only use response sequences (i.e., a series of 0's and 1's showing the correctness of application of a particular skill performed by a particular student) as an input to the detector in the current study.

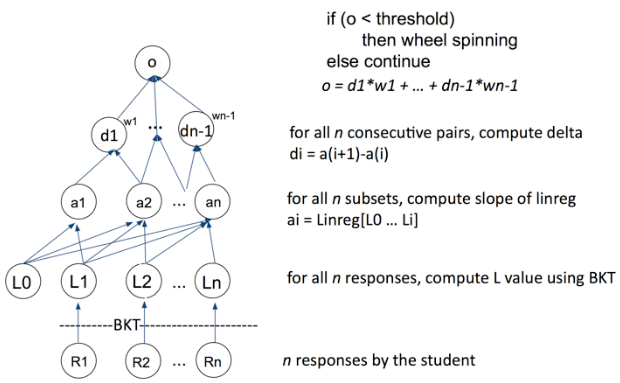
A higher level research question is if we can detect wheel spinning at all: Can we detect wheel-spinning only from the sequence of response accuracy? If so, how accurate the detection is? We hypothesize that if teachers can systematically identify the moment of wheel-spinning only by observing the correctness of student's response, then a neural-network model should be able to learn to detect the moment of wheel-spinning in the same way as teachers do.

2. THE DETECTOR
Our basis for identifying wheel spinning is to analyze the correctness of student responses for a particular skill. We then attempted to test our hypothesis by comparing the predictions of our detector with examples classified by human coders. We asked two human coders to qualify the student's response data to identify wheel-spinning cases based on our coding manual. Table 1 shows a contingency table showing the agreement between two coders. The inter-coder reliability (the Cohen's kappa) on this final coding is 0.90.

Table 1. Inter-coder agreement of wheel-spinning coding

		Coder 2		
		W	C	Total
Coder 1	W	72	13	85
	C	5	752	757
	Total	77	765	842

Having identified the wheel spinning cases, we attempted to train a neural network to learn a latent pattern in a gradual change in a sequence of 1s and 0s, representing the first attempt a student has a step for a certain skill.



The input of the NN-based detector is a *response sequence* (denoted as R_1, R_2, …, R_n in the figure) that shows a chronological record of the correctness of skill application made by the student on a particular skill. Each time a new response is observed (i.e., R_n in the figure), the response sequence is fed into the Bayesian Knowledge Tracer (BKT) to update a predicted mastery level up to the point of the latest response observation (denoted as L_1, L_2, …, L_n).

The first part of our neural network computes the change in the predicted mastery level represented as a slope of a linear regression model with the L value as a dependent variable and the opportunity count (i.e., i in L_i) as an independent variable. The slope of this line represents how gradual the student's learning has been. The second part of the neural network computes the deltas for each of the consecutive slope values. Students who are consistently learning have deltas greater than or equal to 0, because overall the trials that those students make forward progress. However, in the case of wheel spinning, the slopes decrease more often than they increase.

The output from the neural network is a weighted sum of the delta values (in the second hidden layer) representing the likelihood of wheel spinning. We train the neural network to learn weights for each delta values in such a way that the output less than zero indicates a potential of wheel spinning and the smaller the output value the more likely the student would wheel spin. The neural network updates weights using back propagation to converge on a set of weights that minimize the classification error during the training.

3. RESULTS

We used the dataset "Cog Model Discovery Experiment Spring 2010" in the study called "Geometry Cognitive Model Discovery Closing-the-Loop", taken from DataShop[1]. This dataset contained 5385 student-skill responses. Among 5385 student-skill response sequences, there are 2883 response sequences that have more than and equal to 5 responses. We filtered out response sequences with less than 5, because there would not be enough attempts to determine wheel spinning. Out of 2883, there are 842 response sequences that do not reach to the mastery according to BKT (hence potentially wheel spinning). In these 842 response sequences, there are 122 unique students and 44 unique skills included.

For our validation study, we decided to use only student-skill response sequences that had greater than or equal to 10 opportunities, because we were trying to find out the best number of opportunities to predict from 5 to 10. After filtering out instances with less than 10 attempts, we were left with 141 student-skill response sequences. We then randomly dropped one response sequence to have 140 student-skill response sequences for a 10-fold cross-validation. On the 9 folds training data, each of the skill-specific neural networks was trained until it classified training instances with the minimum classification errors. The accuracy of the prediction was computed as an overall average across 10 cross-validations. We computed a precision and recall score for each 10-fold-validation, along with a corresponding F1 score. Figure 3 shows precision, recall, and F1 (which is 2*P*R/(P+R) where P and R shows precision and recall respectively) scores for N = 5 to 10.

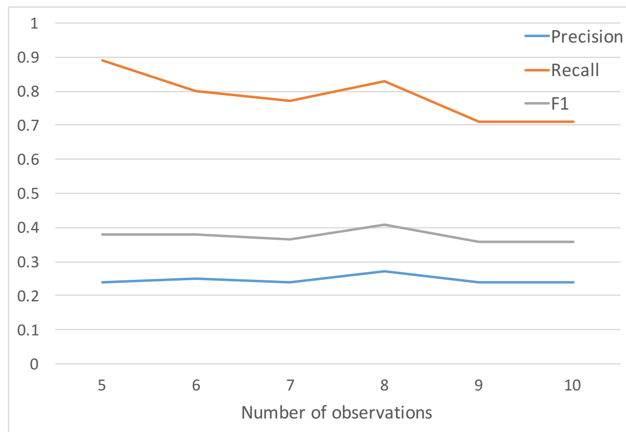


Figure 1. The precision, recall, and F1 scores computed on the first N response observations.

ACKNOWLEDGMENTS

The research reported in this paper has been supported by National Science Foundation Award No. 1418244.

REFERENCES

[1] BECK, J.E. and GONG, Y., 2013. Wheel-Spinning: Students Who Fail to Master a Skill. In *Artificial Intelligence in Education*, H.C. LANE, K. YACEF, J. MOSTOW and P. PAVLIK Eds. Springer Berlin Heidelberg, 431-440. DOI= http://dx.doi.org/10.1007/978-3-642-39112-5_44.

[2] CORBETT, A.T. and ANDERSON, J.R., 1995. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User Adapted Interaction 4*, 4, 253-278.

[3] RITTER, S., ANDERSON, J.R., KOEDINGER, K.R., and CORBETT, A., 2007. Cognitive tutor: Applied research in mathematics education. *Psychonomic bulletin & review 14*, 2, 249-255.

[4] VANLEHN, K., 2006. The Behavior of Tutoring Systems. *International Journal of Artificial Intelligence in Education 16*.

[1] https://pslcdatashop.web.cmu.edu