Early wheel-spinning detection in student performance on cognitive tutors using an ensemble model

Seoyeon Park
Texas A&M University
pseoyeon5@tamu.edu

Noboru Matsuda
Texas A&M University
noboru.matsuda@tamu.edu

Wheel-spinning is a phenomenon in which a student has spent a considerable amount of time practicing a skill without making progress (Beck & Gong, 2013). Several wheel-spinning detectors have been developed using student performance, skills, and other variables with diverse techniques. However, these existing models tend to show low recall rates, indicating the lack of wheel-spinning detecting power. Moreover, it is hard to detect wheel-spinning cases early, before students give up repetitive problem-solving. Thus, the purpose of this study is to build a more simplified and fast wheel-spinning detector using variables based on student response sequences. Results showed that the generic model with logistic regression shows higher accuracy than the existing models. In addition, the upgraded model applying the gradient boosted tree shows not only higher accuracy, but also higher recall rate. This indicates that we can detect wheel-spinning cases on the 5th opportunity with 72% accuracy or the 6th opportunity with 75% accuracy, which is much higher and faster than the existing detectors.

1. INTRODUCTION

Cognitive tutors actively incorporate the mastery learning model as a theoretical basis in their design (Gong, Wang, & Beck, 2016). Multiple theories regarding cognitive tutors have been derived and developed from the mastery learning model. The theory of "learning by doing" supported repeating the same action as practice in order to improve learners' skills (Schank, Berman, & Macpherson, 1999). Anderson (1996) also suggested the ACT-R theory that solving problems repetitively would make procedural knowledge stronger. These models strengthen the concept that practice makes perfect, justifying that most cognitive tutors provide students enough practice problems for their mastery of skills.

However, several researchers have pointed out the limitations of cognitive tutors based on the mastery learning model. Instead of ensuring an efficient number of practice opportunities, cognitive tutors could waste students' learning time and keep them from mastering skills by giving redundant assessments (Cen, Koedinger, & Junker, 2007; Baker, Gowda, & Corbett, 2011). It could also threaten the motivation of both students who already achieved mastery level and others who are stuck without progress, which causes a serious early course termination problem (Seymour & Hewitt, 1997; Watkins & Mazur., 2013)

Beck and Gong (2013) found that cognitive tutors kept failing to detect students who spent a considerable amount of time practicing a skill without achieving mastery and labelled this phenomenon as ‘wheel-spinning’, referring to a car stuck in mud or snow. They found that this phenomenon can be universal throughout cognitive tutors. It suggests, that not all students could eventually master skills within cognitive tutors despite spending a substantial amount of time and effort. Rather, cognitive tutors could hinder some students’ learning and frustrate them, which would discourage them to learn further. Therefore, it is crucial for cognitive tutors to have an effective and reliable detector which can detect students who will wheel-spin without mastery as soon as possible.

Recent studies on wheel-spinning are actively discovering a practical and scalable wheel-spinning detector that can be embedded into cognitive tutors. Beck and Gong (2015) suggested a generic model to predict who will fail to master a skill, employing three aspects: student in-tutor performance on the skill, the seriousness of the learner, and general factors. Student in-tutor performance on the skill represents students’ extent of understanding the skill including the number of correct responses and hint use. The seriousness of the learner across skills is composed of features regarding response speed and the number of consecutive hints. General factors indicates the number of prior problems that students have practiced and the skill difficulty. Matsuda, Chandrasekaran, and Stamper (2016) tried to build more simplified wheel spinning detectors by the combination of the probability of mastery, based on Bayesian knowledge tracing, and a neural-network model.

Both detectors showed a pretty high precision around 70~79%, but relatively low recall rates around 25~50%. This indicates that these models’ abilities to detect wheel-spinning students are insufficient, and it is highly likely to miss wheel-spinning students using these models. What if we want to determine whether a student will wheel-spin on his/her 10th opportunity in the middle of students’ using cognitive tutors, when we only have limited information? Even though wheel-spinning students can be diagnosed after a relatively long period of time, over 10 or 15 practice opportunities by definition, a wheel-spin detector in cognitive tutors should be able to distinguish those students with a high possibility to wheel-spin in advance before they actually wheel-spin. This kind of detector is especially necessary in that the majority of “indeterminate” students, who did not practice on enough opportunities to define their mastery, did not practice on more than five problems (Beck & Gong, 2013; 2016). This indicates that cognitive tutors need a wheel-spin detector which can detect wheel spinning as early as the 5th or 6th opportunity, before students give up on a repetitive problem set. This would be a fundamental intention and practical use of adopting a wheel-spin detector to cognitive tutors.

The goal of this study is to develop a more simple and scalable wheel-spinning detector for a cognitive tutor that can distinguish students who have a high possibility to wheel-spin as quickly as possible. The specific research questions are as follows:

1. How accurately can wheel-spinning be caught with the wheel-spinning detector whose features are primarily dependent on the student responses on a skill/ problem type?
2. How early can we detect wheel-spinning with this wheel-spinning detector?

The features for the wheel-spinning model in this study are all from the sequence of student responses in order to guarantee its simplicity and scalability. We made six features: first, we created “M/W on each opportunity”, based on whether a student has three consecutive correct answers in his/her first attempt (M, meaning mastery) or not (W, meaning wheel-spinning) by each opportunity (3rd, 4th, 5th...9th opportunity). We hypothesize that we might be able to predict students’ eventual wheel-spinning, using the fact that students show mastery or wheel-spinning on each opportunity (3rd, 4th, 5th...9th). For example, when we predicted wheel-spinning using "M/W on the 4th opportunity", we used whether a student failed (W) to have three correct responses in a row within 4 opportunities or not (M) as one of the significant features. Then, we also generated the average probability of correct first attempt per student, skill, student-skill pair, problem type, and student-problem type pair respectively. We hypothesize that a wheel-spin detector, which is primarily dependent on student responses on a skill/ problem type, can be an adaptive wheel-spin detector due to its simplicity and flexibility on how students respond to the problem. In this study, two models are developed as wheel-spinning detectors that apply these features. One is a generic model using logistic regression and the other is labelled as the upgraded wheel-spinning detector which employs a gradient boosted decision tree method.

2. DATA PREPROCESSING

We used an existing dataset from DataShop, entitled 'Cog Model Discovery Experiment Spring 2010' in the 'Geometry Cognitive Model Discovery Closing-the-Loop study' project. There are 49 skills forming 45,597 observations done by 123 students in the 'KTracedSkills' model in this data. This dataset contained 5,279 student-skill pairs. Mastery of a skill in this study is defined as three consecutive correct responses on one's first attempt within 10 practice opportunities, same as Beck and Gong (2013). We filtered out "indeterminate" students, who did not practice on enough opportunities, which is 10 opportunities in this study, for us to define their mastery (Beck & Gong, 2015). This dataset had 1,764 indeterminate student-skill pairs, which accounted for around 33% of the dataset. After removing indeterminate student-skill pairs, this dataset came to contain 3,515 student-skill pairs and 36,887 observations with 123 students. The dependent variable is whether a student master (M) or wheel-spin (W) on a skill within 10 opportunities based on the response sequences of each student-skill pair. If students have three consecutive correct responses on their first attempt on a skill in the limit of 10 opportunities, they are coded as mastery (M) of a skill. If not, they are categorized as wheel-spinning (W) on a skill.

3. FEATURE ENGINEERING

In order to meet our research goal to realize a simplified but reflective wheel-spin detector, our feature engineering focused on utilizing sequences of student responses on their first attempts. Matsuda, Chandrasekaran, and Stamper (2016) used only student-skill response sequences to build a simplified wheel-spin detector and to find out the best number of opportunities to predict between 5 and 10 as well. This model showed greater accuracy, but much less recall rate (around 25%) than that of Beck and Gong (around 50%). We built a wheel spin detector by developing six features: (1) M/W on each opportunity, (2) average probability of making the first attempt on a step correctly per student, skill, student-skill pair, problem type, and student-problem type pair respectively. All features were calculated using information up to each opportunity. For example, 'ID_Average_FirstAttempt_OPP5' means the average correct answer rate of each student across skills throughout 5 opportunities.

3.1. M/W ON EACH OPPORTUNITY

We categorized M/W on each opportunity based on the sequences of responses of student-skill pairs. This categorization needs at least 3 responses in a row so we could make M/W from the 3rd to 9th opportunity. This feature was invented to predict wheel-spinning on the 10th opportunity, employing whether students master a skill on the 3rd, 4th, … 9th opportunity or not.

The relationship between this feature and the dependent variable can have four options. First, if a student achieves mastery (M) on a skill on the *n*th opportunity ($3 \leq n \leq 9$), then the predicted output for the 10th opportunity is supposed to be mastery (M). Second, even though a student shows mastery (M) on a skill on the *n*th opportunity, the model can predict him/her to wheel-spin (W) on the 10th opportunity. This second case can be regarded as a negative example of this feature. Our generic wheel-spinning detector using logistic regression did not have any negative examples throughout all opportunities. The upgraded wheel-spinning detector with the gradient boosted decision tree model showed 4 negative examples among 36,887 observations when only on the 3rd opportunity. Third, even though a student is wheel-spinning (W) on a skill on his/her *n*th opportunity, he/she can be predicted to master (M) the skill on the 10th opportunity. Lastly, a student who is wheel-spinning (W) on a skill on his/her *n*th opportunity can be predicted to end up wheel-spinning on the 10th opportunity. If M/W during early phases (opp3-opp5) has detecting power of predicting eventual mastery or wheel-spinning on the 10th opportunity, this would help us to build the detector that can detect wheel-spinning as fast as possible.

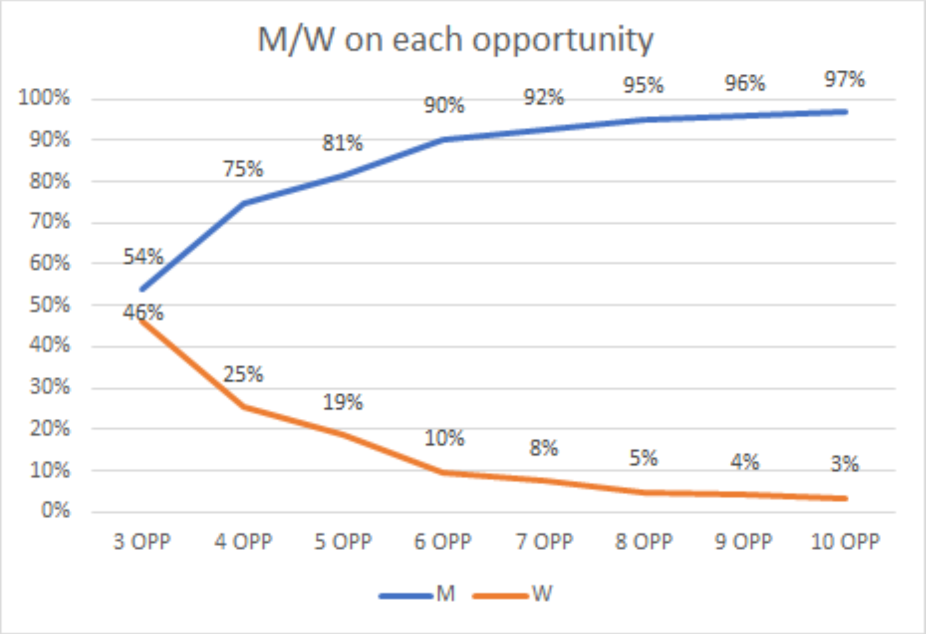

Figure 1. The proportion of M/W on each opportunity

Figure 1 shows how the proportion of M/W changes on each opportunity. M/W on the 3rd opportunity indicates that about 54% of student-skill pairs are mastered on the first attempt. The percentage of mastery goes up and that of wheel-spinning goes down clearly as practice opportunity increases. M/W on the 10th opportunity is the dependent variable that we need to predict, and only 3% of student-skill pairs failed to master within the 10th opportunity.

However, considering that we removed 33% of the dataset associating with 'indeterminate' students who did not try up to 10 opportunities, the percentage of wheel-spinning can be 37% in its maximum level.

3.2. AVERAGE PROBABILITIES OF CORRECT FIRST ATTEMPTS PER STUDENT, KC, AND STUDENT-KC PAIR

We constructed the average probabilities of correct first attempts per student, skill (KC), and student-skill pair respectively. These variables can be analyzed to provide us an insight into how we can evaluate student responses further. We arranged student responses by student, skill (KC), and student-skill pair separately in chronological order (Table 1). Then, we calculated the average rate of correct responses in the first attempts of each category.

Table 1. The process for calculating ID, KC, IDKC_Average_First Attempt.
Note. 'Opp' means 'Opportunity'. Student 'S1' has 0.67 ID_Average_First Attempt on his/her 3rd opportunity {(0+1+1)/3}. Skill 'A' has 0.67 KC_Average_First Attempt on its 3rd opportunity {{(0+1+1)/3}. IDKC 'S1A' has 0.5 IDKC_Average_First Attempt on its 2nd opportunity{(0+1)/2}.

ID (Student)	Opp for ID	KC (Skill)	Opp for KC	IDKC (Student-skill pair)	Opp for IDKC	First Attempt Correct (Correct :1/ incorrect:0)
S1	1	A	1	S1A	1	0
S1	2	B	1	S1B	1	1
S1	3	A	2	S1A	2	1
S2	1	B	2	S2B	1	0
S2	2	A	3	S2A	1	1

ID_Average_First Attempt on each opportunity: This variable indicates the average correct answer rate of a student across skills throughout each opportunity. This can represent students' personal problem-solving ability.
KC_Average_First Attempt on each opportunity: This variable can represent the difficulty of each skill. This is calculated as the average correct answer rate of a skill across all students who practiced the skill.
IDKC_Average_First Attempt on each opportunity: This variable illustrates the average correct response rate of a student on a skill at each opportunity.

3.3. AVERAGE CORRECT FIRST ATTEMPT PER PROBLEM TYPE AND STUDENT-PROBLEM TYPE PAIR

Along with the variables mentioned above, we introduce a new concept, 'Problem Type' in our feature engineering. 'Problem Type (PT)' is generated to measure students' general math competency in a different dimension from students' personal problem-solving ability (ID) and the difficulty of each skill (KC). Variables formed on PT can provide us additional information regarding how we can investigate general math competency in student modelling and how it affect students' mastery. According to the research on variance explained by student, skill and the combination of them across constructs (Beck, Ostrow, & Wang, 2016), student variance explains 17% of wheel-spinning on average. Skill variance attributed to 15% of wheel-spinning on average. We hypothesize that it might be not enough to explain the student model with student and skill variance. Thus, we adopted 'Problem Type', which can offer another granularity of knowledge in analyzing students' learning in cognitive tutors.

Table 2. The process for creating Problem Type (PT).
Note. The example problem (ac-arrow-ca-rt-1-p3) is coded as 'rectangle_triangle_area' problem type, by extracting core words from 'step name' and 'KC' that compose the problem. We marked 'rectangle' as underlined red, 'triangle' as italicized blue, and 'area' as bold green.

Problem Name	Step Name	KC(KTracedskills)	Problem Type (PT)
ac-arrow-ca-rt-1-p3	Q1-**area**-operation	Find added **area**	rectangle_*triangle*_**area**
	rectangle-1-Q1-**area**	Find individual **area** Find individual area in context	
	rectangle-1-Q1-base	Enter given measurement	
	rectangle-1-Q1-height	Enter given measurement	
	triangle-2-Q1-**area**	Find individual **area** Find individual area in context	
	triangle-2-Q1-base	Enter given measurement	
	triangle-2-Q1-height	Enter given measurement	

Table 2 shows the process for building PT. We arranged the steps and skills based on how they comprise a problem, extracted representative words from 'Step Name' or 'KC', and created PT for each problem. The example problem (ac-arrow-ca-rt-1-p3) in Table 2 is assumed to be relevant to 'the area of rectangle and triangle' based on its 'Step Name' and 'KC'. This concept is not robust and the process for making PT is not the only way to resolve the issue. We are further exploring how we are able to grasp students' learning in a cognitive tutor better.

Table 3. The process for calculating PT, IDPT_Average_First Attempt
Note. 'Circle_rectangle' problem type shows 0.5 {(0+1)/2} PT_Average_FirstAttempt on its 2nd opportunity. IDPT 'S2Triangle_circle' has 0.5 {(0+1)/2} IDPT_Average_First Attempt on its 2nd opportunity.

ID (Student)	PT (Problem Type)	Opp for PT	IDPT (Student-PT pair)	Opp for IDPT	First Attempt Correct (Correct :1/ Incorrect:0)
S1	Circle_rectangle	1	S1Circle_rectangle	1	0
S1	Triangle_circle	1	S1Triangle_circle	1	1
S1	Circle_rectangle	2	S1Circle_rectangle	2	1
S2	Triangle-circle	2	S2Triangle_circle	1	0
S2	Triangle_circle	3	S2Triangle_circle	2	1

PT_Average_First Attempt on each opportunity: This variable implies the difficulty of each problem type. This is calculated as the average correct answer rate of a problem type across all students who practiced the type.
IDPT_Average_First Attempt on each opportunity: This variable illustrates the average correct response rate of a student on a problem type by each opportunity.

Principal-components analysis was conducted to find how these five numerical features explain the dependent variable. The first component (PC1) accounted for 63%, the second component (PC2) for 21%, and the third (PC3) for about 10% of total variance. It is found that PC1 is strongly correlated with IDKC_Ave_First Attempt. PC2 increases with increasing IDPT_Ave_First Attempt. We can assume that how well students do on both KC and PT can be strong predictors for detecting wheel-spinning cases.

Table 4. The correlations between the principal components and variables.

Variable	PC1	PC2	PC3
PT_Ave_First Attempt	0.137	0.291	-0.504
IDPT_Ave_First Attempt	0.340	0.849	-0.024
IDKC_Ave_First Attempt	0.799	-0.235	0.455
KC_Ave_First Attempt	0.467	-0.333	-0.670
ID_Ave_First Attempt	0.091	0.169	0.298

4. A GENERIC WHEEL-SPINNING DETECTOR WITH LOGISTIC REGRESSION

We trained separate logistic regression models with a ten-fold cross-validation at each opportunity to verify that our features can make a generic model with comparable performance. The dependent variable is whether a student achieves mastery (M) or wheel-spins (W) on a skill. Tests for multicollinearity indicated that a very low level of multicollinearity was present (VIF= 1.44 for ID_Ave_First Attempt, VIF=2.46 for KC_Ave_First Attempt, VIF=2.67 for

IDKC_Ave_First Attempt, VIF=1.81 for PT_Ave_First Attempt, and VIF=2.21 for IDPT_Ave_First Attempt). Detailed estimates are omitted due to the limited space.

4.1. MODEL ACCURACY AND MISCLASSIFICATIONS OF OUR GENERIC MODEL

Our generic wheel-spinning model shows high accuracy throughout opportunities. Percent Correct (the percent of correct model predictions) and AUC (the area under curve of the ROC) is used to measure model accuracy at each opportunity. We can clearly see that both Percent Correct and AUC increase by the increase of practice opportunity. Considering that the accuracy of the wheel-spinning model of Beck and Gong (2015) was less than 90% (AUC is under 0.9), this model consistently shows good performance with less number of variables throughout opportunities.

Table 5. Model Accuracy of the wheel-spinning detector

	opp3	opp4	opp5	opp6	opp7	opp8	opp9
Percent Correct (%)	94.85	95.19	95.41	96.28	96.55	98.2	98.96
AUC	0.926	0.944	0.96	0.977	0.986	0.995	0.998

As our main interest is how well this model can detect wheel-spinning cases as rapidly as possible, we examine the precision and recall rate of this model at each opportunity. Precision illustrates the proportion of actual wheel-spinning cases among the predicted ones. Recall means the percentage of correct wheel-spinning prediction by this model. Average precision rate is 72% and recall rate is 57% in this model. Table 6 shows the trend of precision and recall rate for this data on wheel-spinning by each opportunity. This result looks similar to the precision and recall rate of ASSISTments (Beck & Gong, 2015). Precision rate remains relatively constant and increases slightly by opportunity. Recall rate has a more dynamic increase as opportunity increases. In the early phases (opp3 - opp5), recall rate is increasing but it is still pretty low, less than 40%, due to the cold start problem. The precision and recall rate at the 6th opportunity is similar to the average precision and recall rate of this model.

However, the recall rate is still around 55%, which means we might fail to detect half of the wheel-spinning cases if we determine wheel-spinning at the 6th opportunity with this model. We are able to conclude wheel-spinning cases at the 7th opportunity when both precision and recall rate make a balance around 70%, which is also corresponding to the result of Beck and Gong's generic model. All these values, accuracy, precision and recall rate, reveal that our model has good performance with less variables than the existing models.

Table 6. Precision and recall scores of the detector with a specific number of observations

	opp3	opp4	opp5	opp6	opp7	opp8	opp9
Precision (%)	61.53	67.06	66.61	72.08	70.74	78.4	84.39
Recall (%)	21.45	27.6	36.22	54.62	65.57	93.7	100

5. THE UPGRADED WHEEL-SPINNING DETECTOR WITH GRADIENT BOOSTED DECISION TREE

We found that our generic model, using logistic regression, shows somewhat similar but improved results in its accuracy, precision and recall rate when compared to other models. In order to build a wheel-spinning detector that can detect wheel-spinning cases more rapidly, we explored other machine learning techniques, especially for addressing the low recall rate. We trained gradient boosted decision trees with a ten-fold cross validation by each opportunity.

A gradient boosted decision tree model is an ensemble of decision tree models. The gradient descent boosting paradigm is developed for additive expansions by Friedman (2001). The gradient boosting of regression trees produces highly robust procedures for both regression and classification (Friedman, 2001). The boosting algorithm indicates generating a series of classifiers (Freund & Schapire, 1996). In the series of classifiers, new classifiers are created to primarily depend on incorrectly predicted examples in the previous classifiers which make the previous ensemble performance poor (Maclin & Opitz, 1997). Repeating this procedure for many rounds enables a dataset to have a different distribution or weights over the training examples, and a single prediction rule with much higher accuracy is created by the combination of weak algorithms (Schapire, 2003). We used the default setting in Rapidminer[1] with 0.1 learning rate, 5 maximal depth, and 20 number of trees, due to the dependence of this model on the setting.

5.1. MODEL ACCURACY AND MISCLASSIFICATIONS OF THE UPGRADED WHEEL-SPINNING DETECTOR

The upgraded wheel-spinning model shows high accuracy throughout opportunities as well. The values of both Percent Correct and AUC are slightly higher than those of our generic detector. However, the precision and recall rate at each opportunity differs considerably from that of the generic model.

Table 7. Model Accuracy of the upgraded wheel-spinning detector

	opp3	opp4	opp5	opp6	opp7	opp8	opp9
Percent Correct (%)	94.03	94.75	95.3	96.78	97.68	98.84	99.48
AUC	0.936	0.961	0.973	0.986	0.993	0.998	1

Average precision rate is 69% and recall rate is 77% in this model. The average precision rate is slightly lower and recall rate is 20% higher compared to that of the generic model. Table 8 indicates how the precision and recall rate for this data on wheel-spinning changes by each opportunity. Both the precision and recall rate increases as opportunity increases. Interestingly, unlike the values of the generic model, recall rate is higher than precision rate throughout all opportunities. Applying this model, the recall rate starts over 50% on its 3rd opportunity and over 70% on its 5th opportunity. If we want to point out the opportunity on which the precision and recall rate shows balance, we might be able to pick the

[1] https://rapidminer.com

6th opportunity where its precision rate is about 70% with high recall rate, about 75%. In terms of the goal of this wheel-spinning detector, which is detecting wheel-spinning cases as early as possible, this upgraded wheel-spinning detector can be a powerful tool to distinguish those cases in the early phase of using cognitive tutors.

Table 8. Model misclassifications of the upgraded wheel-spinning detector

	opp3	opp4	opp5	opp6	opp7	opp8	opp9
Precision(%)	47.05	52.59	56.25	69.73	77.98	85.14	94.66
Recall(%)	53.22	63.83	72.15	74.96	81.65	96.03	96.22

5.2. COMPARING THE PRECISION AND RECALL RATE OF TWO MODELS

Figure 2 compares the precision rate and recall rate at each opportunity between the generic model and the upgraded model. The blue line is for the generic model, which used logistic regression, and the orange line is indicating the upgraded model using the gradient boosted decision tree. In the precision rate, the generic model shows higher performance up to the 6th opportunity. After the 7th opportunity, the upgraded model overtakes. In the recall rate, the upgraded model displays a much higher rate up to the 8th opportunity. Even though the difference of two models' recall rate shrinks by each opportunity, we can assume that the upgraded model has a much higher recall rate, especially in the early phases (opp 3-opp 5). This indicates that detecting wheel-spinning cases with the gradient boosted classification tree model can increase the probability on detecting wheel-spinning.

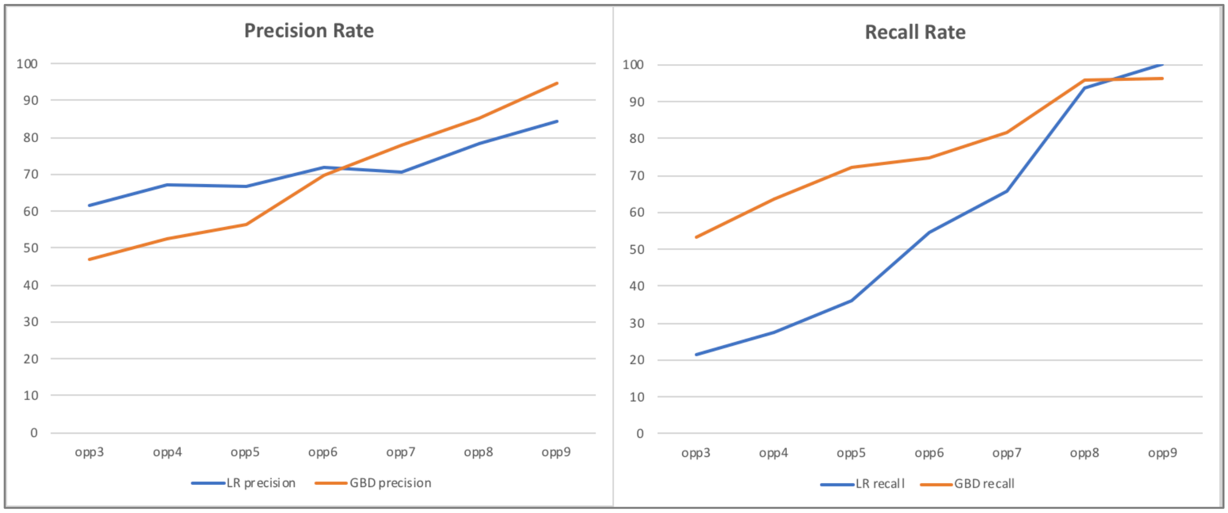

Figure 2. Comparison between the precision and recall rate of two models

6. DISCUSSION, CONTRIBUTION, AND CONCLUSION

This study sought to develop a more simple and scalable wheel-spinning detector for a cognitive tutor that can distinguish wheel-spinning cases as quickly as possible. The focus of the present work is to build the wheel-spinning detector with enhanced accuracy and speed by applying six features based on the sequence of student responses. We have some important findings in this work. First, we found that multidimensional average correct first attempt can be sufficient in predicting wheel-spinning cases. We considered students' response sequences from various angles, which combined levels of student, skills, and problem type. This combination of features shows great performance in accuracy across different machine learning techniques, logistic regression and gradient boosted decision trees in this study. This is a significant finding as it implies that an adaptive wheel-spin detector can be made by only using student responses, which emphasizes its simplicity and scalability. This is somewhat even intuitive for teachers to identify the wheel-spinning moment by solely monitoring student responses on the cognitive tutor.

Second, we applied an ensemble machine learning technique to build a more improved wheel-spinning detector. Our upgraded wheel-spinning detector contributes to enhance the low recall rate, which was one of the main issues of existing wheel-spinning detectors. The result suggested that its average recall rate is 77% with good performance in accuracy and precision rate on average. Considering that the average recall rate is about 50% or lower in other existing models, including our generic model using logistic regression, this upgraded model implementing the ensemble techniques can be regarded to strengthen the wheel-spinning detecting power. Moreover, this model can catch wheel-spinning cases with over 70% accuracy on its 5th opportunity. This result showed that this model can detect wheel-spinning in the early phase of using cognitive tutors. Thus, our upgraded model augments not only the overall accuracy in predicting wheel-spinning moments but also the speed of detecting those cases.

The present work has some limitations. First, this model needs to be tested in various types other datasets in addition to DataShop, which was our data source for the current study. Applying this model to the actual data in MOOCs might also help to increase the generality of this model. Secondly, another limitation is the measure of mastery that were used in this work. We defined mastery and wheel-spinning as Beck and Gong (2013) did, which is whether a student has three consecutive correct answers in his/her first attempts. Considering not all systems define mastery as we did, this model would not be suitable for detecting wheel-spinning cases based on other definitions of mastery and wheel-spinning, such as using Bayesian Knowledge Tracing or five consecutive correct answers in a row.

Needless to say, one of the intriguing future works might be to find the effective intervention for wheel-spinning cases. What can we do when we detect wheel-spinning on the 5th attempt? Or the 7th? Another suggestion for future work is to develop a wheel-spinning detector for other systems using a different definition of mastery. Would this model, which is mainly using student responses as its features, also be effective in detecting wheel-spinning in other systems with a different definition of mastery? Finally, we can deepen our knowledge further regarding why students actually wheel-spin while learning in cognitive tutors.

7. ACKNOWLEDGEMENT

We would like to thank Carnegie Learning, Inc., for providing the Cognitive Tutor data supporting this study. This study is supported by NSF grants No. 1623702 and 1644430.

REFERENCES

ANDERSON, J. R. 1996. ACT: A simple theory of complex cognition. *American Psychologist, 51*(4), 355.

BAKER, R. S., GOWDA, S. M., AND CORBETT, A. T. 2011. Towards predicting future transfer of learning. In *International Conference on Artificial Intelligence in Education*. Springer, Berlin, Heidelberg, 23-30.

BECK, J. E., AND GONG, Y. 2013. Wheel-spinning: Students who fail to master a skill. In *International Conference on Artificial Intelligence in Education.* Springer, Berlin, Heidelberg, 431-440.

BECK, J., OSTROW, K. AND WANG, Y. 2016. Students vs. Skills: Partitioning Variance Explained in Learner Models. In *The 9th International Conference on Educational Data Mining*. ACM

CEN, H., KOEDINGER, K., AND JUNKER, B. 2006. Learning factors analysis–a general method for cognitive model evaluation and improvement. In *International Conference on Intelligent Tutoring Systems*. Springer, Berlin, Heidelberg, 164-175.

FREUND, Y., AND SCHAPIRE, R. E. 1996. Experiments with a new boosting algorithm. In *Icml. 96*, 148-156.

FRIEDMAN, J. H. 2001. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, 1189-1232.

GONG, Y., AND BECK, J. E. 2015. Towards detecting wheel-spinning: Future failure in mastery learning. In *Proceedings of the Second (2015) ACM Conference on Learning@ Scale*. ACM, 67-74.

GONG, Y., WANG, Y., AND BECK, J. 2016. How long must we spin our wheels? Analysis of student time and classifier inaccuracy. *Student modeling from different aspects*, 32-38.

MACLIN, R., AND OPITZ, D. 1997. An empirical evaluation of bagging and boosting. *AAAI/IAAI*, 546-551.

MATSUDA, N., CHANDRASEKARAN, S., AND STAMPER, J. C. 2016. How quickly can wheel spinning be detected?. In *EDM.* 607-608.

SCHANK, R. C., BERMAN, T. R., AND MACPHERSON, K. A. 1999. Learning by doing. *Instructional-design theories and models: A new paradigm of instructional theory, 2,* 161-181.

SCHAPIRE, R. E. 2003. The boosting approach to machine learning: An overview. In *Nonlinear estimation and classification*. Springer, New York, NY, 149-171.

SEYMOUR, E., AND HEWITT, N. M. 1997. *Talking about leaving: Why undergraduates leave the sciences*. Boulder, CO: Westview.

STAMPER, J., AND RITTER, S. 2010. Cog Model Discovery Experiment Spring 2010. Dataset 392 in DataShop. Retrieved from https://pslcdatashop.web.cmu.edu/DatasetInfo?datasetId=392.

RITTER, S., ANDERSON, J.R., KOEDINGER, K.R., AND Corbett, A. 2007. The Cognitive Tutor: Applied research in mathematics education. *Psychonomics Bulletin & Review, 14*(2), 249-255.

WATKINS, J., AND MAZUR, E. 2013. Retaining students in science, technology, engineering, and mathematics (STEM) majors. *Journal of College Science Teaching, 42*(5), 36-41.