Predicting Young Students' Self-Evaluation Deficits Through Their Activity Traces

Thomas Sergent
Sorbonne Université, CNRS,
LIP6, F-75005 Paris, France
Lalilo, Paris, France
thomas.sergent@lip6.fr

Morgane Daniel
Lalilo, Paris, France
morgane@lalilo.com

François Bouchet,
Thibault Carron
Sorbonne Université, CNRS,
LIP6, F-75005 Paris, France
{francois.bouchet,
thibault.carron}@lip6.fr

ABSTRACT

Self-evaluation is a key self-regulatory process that can already be mastered by young children. In order to assess self-evaluation skills of children, we introduced a random prompt asked randomly after 1 out of 15 exercises into a literacy web-application for primary school student, in order to evaluate the perceived difficulty [Too easy, Good, Too difficult] of the exercise they just solved. Comparing students' actual performance with their responses to this prompt can provide information about their ability to self-evaluate, and thus detect students who could improve their self-evaluation skills. We collected more than 1,000,000 responses from 300,000 students and used these data as well as performance data on each question of each exercise to predict a student's response to the next prompt, thereby estimating how likely they are to having a self-evaluation deficit. The results show (a) that a student's past responses to self-evaluation statements impacts the quality of future predictions (b) that the impact of past responses - vs their current performance - is greater when the student has low capacity for self-evaluation (c) that including older student data (answers from several sessions ago) helps in improving the accuracy of the prediction. These results pave the way (1) for adaptive polling by identifying when the model is unreliable, giving them the statement then instead of randomly, (2) for adaptive feedback, by knowing the students the most likely to show a deficit, to provide remediation.

Keywords

Self-Regulated Learning, Primary school, Self-evaluation, Prediction, Remediation, Adaptive polling

1. INTRODUCTION

Improving children's self-regulated learning (SRL) skills is a key component of their academic performance, as self-regulated students generally know better "how to learn", which can have a positive impact in all disciplines [22]. A key SRL process is self-evaluation [17], which is a skill already developed in children as young as 5 years old [19]. It is therefore a particularly interesting SRL aspect to target when working with young children. The most reliable way to assess SRL deficits is through direct questions to the students [1], but constant prompting can lead to an overall degraded perception of the learning environment [3] and it is therefore critical to limit prompting to the minimum. Hence we are interested here in trying to predict students' answers to assessment on perceived difficulty which are used to assess students' tendencies to overevaluate or underevaluate. The second aspect we investigate is relative to the features that are the most relevant for this prediction.

2. RELATED WORK

The EDM community has recognized early on the interest of studying SRL through data mining [21], and previous works have been more particularly interested in detecting SRL behaviors from traces [5], discussion forums [9, 8] or proxy behaviors such as gaming the system [4], explaining such behaviors with sequence mining [20, 2], mixture models (for procrastination, a proxy of SRL) [13] or coherence analysis [18]. Other works have focused on analyzing the differences in use of SRL strategies [6], providing feedback to encourage them [7] or predicting their use [15]. However, as far as the authors are aware, no previous work has specifically attempted to predict how a student would answer to a question aiming to measure a SRL deficit (self-evaluation or any other), and none of the aforementioned work focused on young children (5-7 years old). It is worth noting that although young children's abilities to use SRL strategies may be more limited than in teenagers, they seem to have comparable monitoring skills [16]. Indeed, recent work on a dashboard supporting SRL in a mathematics software program for 9-10 years old (only slightly older than our targeted students) showed a significant improvement in SRL skills for students in the dashboard group compared to those without the dashboard [12].

3. SELF-EVALUATION ASSESSMENT

3.1 Context

Lalilo is one of the many web applications used by teachers in the classroom to help them implement a differentiated pedagogy. At the beginning of 2021, it is used by 40,000 English and French speaking kindergarten and elementary classes every week to strengthen literacy through series of exercises adapted to the students' level, while providing the teacher with a dashboard to evaluate the students' activities and progress. It is therefore a relevant testing ground for evaluating and then trying to correct some students' SRL deficits. A typical session lasts 20 minutes (on average) with the student performing around 15 short exercises with 3 to 7 questions each. Student activities (e.g. logging in, time spent on an question/exercise, mistakes) are traced and we will focus on students' answers to an exercise, thus we will call **trace** only the answers to this set of questions of the same type within an exercise.

3.2 Data collection

To assess some aspects of students' SRL skills, we introduced a random prompt is once every fifteen exercises when a student finishes an exercise (i.e. on average once per typical learning session). This prompt includes a **perceived difficulty** statement asking the student *"How difficult was this exercise for you?"* with 3 possible answers: "Too hard", "Just-right", "Too easy"). Comparing the answer to the perceived difficulty statement with the real performance aims at measuring the **self-evaluation** ability of the students, i.e. their ability to correctly estimate the difficulty of the questions they just answered. Before introducing the assessments, we checked qualitatively in a classroom using Lalilo that statements were understood by 1st grade students (details not presented here). We collected traces from Kindergarten, 1st grade and 2nd grade classes based in France, Canada and USA learning in French (FR) or English (EN) between January 18 and February 24, 2021 on the Lalilo platform. We kept only the traces for which students had answered to a prompt and further on we call trace the answers to the exercise with the associated answers to the prompt.

4. METHODS

4.1 Dataset

Given the history of a student answers to the perceived difficulty prompt and their performance on the current exercise, we want to predict which answer is the most likely to be given by the student to the next prompt (and thus extrapolate their self-evaluation skills). If the student's performance - which will be defined in the Feature engineering subsection - was "excellent" (resp. "poor") and they answered "Too hard" (resp "Too easy"), they were considered as having an underevaluation (resp. overevaluation) deficit. We filtered out students who had strictly less than 8 traces with answers to prompts so that our model would not overfit on results of students with very few answers, as students with very few answers were overrepresented in our initial dataset. We finally had 424,173 traces with an answer to the perceived difficulty statement from 34,083 students having on average 12 traces with self-evaluation answers (SD = 5.93).

4.2 Feature engineering

We engineered several new features that could have a predicting power in our results.

Basic performance feature. In addition to the trace and student IDs, used for filtering but not for prediction, we extracted for each trace the *answer correctness list*, a boolean vector of a length of 3 to 7 (number of questions per exercise).

Enriched performance features. From the answer correctness list, we extracted 5 additional features: the *good answer count* (i.e. the number of 1s in the vector - the higher the value, the better the student may feel they have succeeded), the *total answer count* (i.e. the length of the vector), *the success rate* (i.e. the ratio between the good answer count and the total answer count) and the *second half success rate* (i.e. the success rate on the last half of a trace - a student with self-evaluation deficit may suffer from a recency bias, influencing positively [resp. negatively] their perception of their performance when answering correctly [resp. incorrectly] to the last questions of the exercise).

Exercise features. We hypothesize that self-evaluation deficits are not uniform across one's knowledge. In particular, a student's deficit may be stronger in some types of exercises or on exercises about a given topic. To assess this impact, we added 5 features that relate to the exercise finished just before the two difficulty statements: *exercise template* (for example a multiple choice question or a word composition exercise), *lesson index* (there are around 1,000 lessons in Lalilo - although they are not entirely linear, the higher the value of this feature, the more advanced the content is; when working on English (resp. French) data, the lesson index FR (resp. EN) is empty), *lesson type* (lessons are organized in a tree structure - lesson type represents the first level category), *lesson subject* (lesson subject represents the second level category in the tree), *language* (English or French).

Previous feedback modalities. In order to help students' in their performance assessment, they are randomly given an audio feedback (such as "In the last exercise, you found 3 correct answers of 5 questions") and/or a visual gauge of as many green ticks and red crosses as they had good and bad answers in the previous exercise. Even when these synthesis exercise-level performance indicators are not there, the students always have an immediate question-level true/false feedback. We have previously shown the positive impact of these two indicators in correcting some self-regulation issues, and we therefore hypothesize that they need to be taken into account when predicting how the student will assess the difficulty. Hence we added two binary features, *gauge* and *audio* which indicate whether these performance feedback were given before the two difficulty statements. A feedback is also provided after answering to the prompt, when students display a self-evaluation deficit (as defined in 3.2), encouraging them to be more confident or warning them to be more careful; we therefore encode this as a third binary feature, *remediation*, indicating whether the student received a feedback the last time the difficulty was assessed.

Self-evaluation deficit lag features. Self-evaluation deficits are expected to be a recurring phenomena in students' answers, i.e. a student who has under/overevaluated themselves a few times is likely to under/overevaluate themselves again in the future. Hence we added 3 lag features for the last 3 perceived difficulty assessments. Moreover, since it is possible that the last 3 assessments were not allowing the student to exhibit a deficit (e.g. a student cannot appear to be overevaluating if their performance is at 100% on the last 3 exercises where they were asked to assess the difficulty), we also added 3 lag features for the last 3 perceived difficulty assessments where the student's performance was equal to the performance on the current exercise. Performance is a categorical value which is worth "poor" if the success rate is below 34%, "excellent" if the success rate is at 100% and "medium" otherwise.

Overall previous self-evaluation deficit. If students are stable over time in their assessment, we expect that taking into account the whole history would have a positive impact on the prediction. We therefore introduced 5 additional features: *self-evaluation answer rank* (the number of times the student has been asked a self-assessment), *number of "Too easy" (resp. "Too hard") answers* (the number of times the student previously answered "Too easy" (resp. "Too hard") to previous assessments), and *"Too easy (resp. Too hard) answers ratio"* (the ratio between the two previous features). Additionally, similarly to what was done for the lag features, we also considered the 5 equivalent features exclusively on previous assessment given after an exercise with a similar performance level.

4.3 Algorithms

We used Catboost [14] and LightGBM [10] to perform the predictions, two gradient boosting algorithms based on Decision Trees whose main assets are: (a) their ability to natively deal with categorical features and (b) their explainability, allowing to study feature importance in each prediction with SHapley Additive exPlanations (also called SHAP values) [11]. They also have recently won Kaggle competitions on a variety of datasets. We used MultiClass as a loss function, set the number of iterations at 200 and kept the other hyperparameters at their default values. As their results were very similar for the global prediction task (cf. Table 1), we used CatBoost model only for the other tasks.

4.4 Analyzing features importance

We first measured the improvements allowed by feature engineering to the global prediction scores using 5-fold cross validation and stratifying by student so that no student traces are both in training and testing folds. For all feature importance measures subsequently described, we created a training and a testing set - also stratified by student - and measured the feature importance in the testing set. We studied the importance of various features in our model using the SHAP package [11]. We also compared the importance of features across the three classes so as to highlight the features that have the most impact on each class specifically.

If students showed a given deficit regularly - we defined a threshold at 50% of "underevaluation" (resp. "overevaluation") over the traces with 100% (resp. below 34%) success rate - we tagged the student as having an "underevaluation (resp. overevaluation) deficit". We then trained our algorithms on a dataset with students tagged as having a deficit and on a dataset with students tagged as having no deficit. Our hypothesis was that feature importance would vary between these two models: the predictions were expected to depend more on past answers than current performance for students tagged with deficits compared to the other students.

Finally, we measured the evolution of the performance of the model depending on the self-evaluation answer rank that is predicted. To do so, we analyze the evolution of Cohen's Kappa coefficient, measuring the quality of our prediction of the perceived difficulty, on a cohort of students of the testing set having answered a given amount of prompts. We computed this coefficient for each self-evaluation answer rank and expected the quality of the prediction to improve as students would be better and better characterized by their features.

5. RESULTS AND DISCUSSION

5.1 Global features importance analysis

Firstly, both Catboost and LightGBM allow to predict with a reasonable overall performance the students' perceived difficulty (cf. right part of Table 1). Secondly we see that all features additions improve the model except the previous feedback ones. Specifically, adding features describing what a student did in the past improves the predictions significantly.

Table 2 ranks the top 10 features with their mean absolute SHAP values. Interestingly, the success rate of the student and the success rate on the second half of the correctness list are only the 5th and 8th ranked features. It means that past information about the previous answers of students to self-evaluation prompts influences more the prediction than their current performance, although they are being asked "How difficult was **this** exercise for you?". Specifically, the last three answers to the perceived difficulty question bear a significant weight in our model's predictions, as well as the global "Too easy" and "Too hard" ratios. As expected, the "Too easy" ratio has a huge importance for the "Too easy" class as has the "Too hard" ratio for the "Too hard" class and both ratio are highly ranked for the "Just-right" class. Indeed, we did not input the "Just-right" ratio as the model can learn it from the combination of "Too hard" and "Too easy" ratio. We can note that the success rate feature is mainly important for the "Too easy" and "Too hard" class, which is logical as an excellent (resp. poor) performance is not likely to lead to a "Too hard" (resp. "Too easy") answer. We can also see on Figure 1, as the "Too hard" ratio is equal to 0, it drives the prediction score of the "Too hard" class downwards while it drives the prediction score of the "Just-right" class upwards. Furthermore, the 3 last answers to the perceived difficulty statement of this student were "Just-right", "Too easy", "Too easy" and their success rate on this trace was 100%; the predictions of the "Too easy" class are therefore also driven upward by these features.

5.2 Features rank from self-evaluation deficits

Figure 2 shows the feature importance rankings for students detected as having or not self-evaluation deficits. Students with deficits consistently choose how to answer to the prompt more based on past answers (in particular the "Too easy/hard" ratios), as opposed to students with no deficits who rely more on the success rate to this exercise, as one should. These results are in line with our hypotheses.

5.3 Predicting power based on answer rank

Figure 3 shows the kappa evolution depending on the number of past self-evaluation assessments. The kappa for the first answer is around 0.13, then quickly climbs around 0.4 for the next four traces; and finally slowly increases until plateauing around 0.6. The kappa of 0.13 for the first answer is consistent with Table 1: at the beginning, Student features are empty and the model can only rely on Trace features. With Trace features only, the model reached a Kappa of 0.1084 which coincides with the kappa value of 0.13 in Figure 3. We can then deduce that the model is more and more able to predict answers to the perceived difficulty statement.

Table 1: Prediction metrics after 200 iterations of the Catboost and LightGBM algorithms depending on the features used. We measure mean and standard deviation (in parenthesis) on 5-fold cross validation.

	Trace features			Student features			
Algorithm	Enriched perf.	Exercise	Feedback	Lag	Overall prev. self-eval.	Accuracy	Kappa
CatBoost	Yes					0.4662 (0.0006)	0.0866 (0.0030)
CatBoost	Yes	Yes				0.4737 (0.0018)	0.1084 (0.0022)
CatBoost	Yes	Yes	Yes			0.4736 (0.0018)	0.1081 (0.0026)
CatBoost	Yes	Yes	Yes	Yes		0.6676 (0.0034)	0.4522 (0.0053)
CatBoost	Yes	Yes	Yes	Yes	Yes	**0.6701** (0.0028)	**0.4575** (0.0042)
LightGBM	Yes	Yes	Yes	Yes	Yes	**0.6706** (0.0034)	**0.4565** (0.0032)

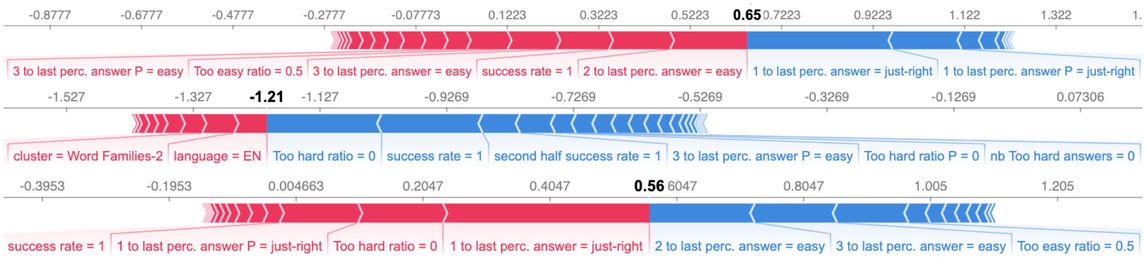

Figure 1: Feature impact in the prediction of each class for a randomly chosen trace in the testing pool. Top: "Too easy" class, middle: "Too hard" class, bottom: "Just-right" class.

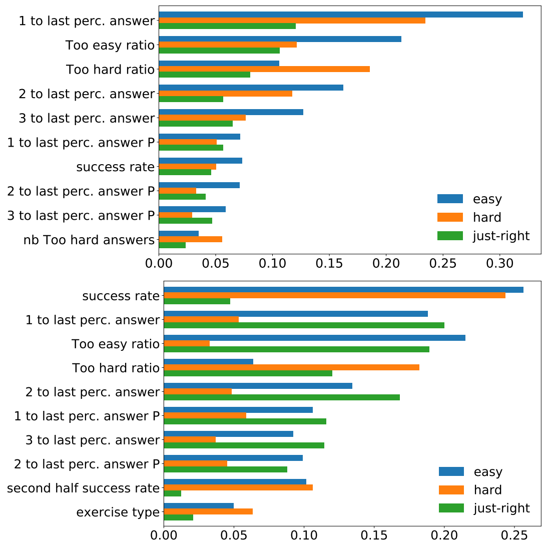

Figure 2: Top 10 features impact for class prediction of students detected as having (top) or not (bottom) a self-evaluation deficit

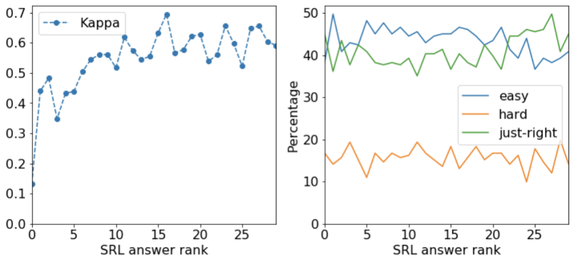

Figure 3: Kappa value and total number of traces of each class in the testing group, based on the self-evaluation answer rank

Table 2: Average feature importance rank by class, sorted by total SHAP value (top 5 in bold)

Features	Too easy	Just-right	Too hard
"Too easy" ratio	**1**	**1**	14
1 to last perc. answer	**2**	**2**	**3**
"Too hard" ratio	6	**4**	**1**
2 to last perc. answer	**3**	**3**	9
success rate	**4**	9	**2**
3 to last perc. answer	**5**	**5**	10
1 to last perc. answer P	7	6	12
second half success rate	8	19	**4**
3 to last perc. answer P	9	7	11
exercise type	10	15	**5**

6. CONCLUSION AND FUTURE WORKS

Using a large volume of trace data from primary school students, we leveraged students' past data to significantly improve the prediction of the answers to future self-evaluation prompts. The results also indicate that the more data we have about a student, the better our predictions are. Using feature engineering, we ranked features by the additional predicting power they provide, and found results consistent with SRL theories (in particular that prediction of answers for well-regulated students depends mostly on their success rate). This paves the way for adaptive polling (as opposed to the current random one), prompting only students likely to display a self-evaluation deficit, allowing us to better target remediation. The main limit of the current work is the specificity of the context: it would be particularly interesting to study the main features used in another context with a different type of students. We are also targeting one of many existing SRL deficits, and expanding research on predicting other deficits to encourage the training of multiple SRL skills seems important as well. Future works also include further feature engineering to refine what features may have more impact than the current ones.

7. REFERENCES

[1] L. Barnard, W. Y. Lan, Y. M. To, V. O. Paton, and S.-L. Lai. Measuring self-regulation in online and blended learning environments. *The Internet and Higher Education*, 12(1):1–6, Jan. 2009.

[2] F. Bouchet, J. M. Harley, and R. Azevedo. Impact of Different Pedagogical Agents' Adaptive Self-Regulated Prompting Strategies on Learning with MetaTutor. In *Proc. of the 16th Conf. on Artificial Intelligence in Education (AIED 2013)*, volume 7926 of *Lecture Notes in Computer Science*, pages 815–819, Memphis, TN, July 2013. Springer Berlin Heidelberg.

[3] F. Bouchet, J. M. Harley, and R. Azevedo. Evaluating Adaptive Pedagogical Agents' Prompting Strategies Effect on Students' Emotions. In *Intelligent Tutoring Systems: 14th Int. Conf.*, volume 10858 of *LNCS*, pages 33–43, Montreal, QC, Canada, June 2018. Springer.

[4] S. Dang and K. Koedinger. Exploring the Link between Motivations and Gaming. In *Proc. of the 12th Int. Conf. of Educ. Data Mining*, pages 276–281, 2019.

[5] N. Diana, M. Eagle, J. C. Stamper, and K. R. Koedinger. Extracting Measures of Active Learning and Student Self-Regulated Learning Strategies from MOOC Data. In *Proc. of the 9th Int. Conf. on Educ. Data Mining*, pages 583–584, Raleigh, NC, USA, 2016.

[6] E. Farhana, T. Rutherford, and C. F. Lynch. Investigating Relations between Self-Regulated Reading Behaviors and Science Question Difficulty. In *Proc. of the 13th Int. Conf. on Educ. Data Mining*, pages 395–402, Ifrane, Marocco, 2020.

[7] J. Feild. Improving Student Performance Using Nudge Analytics. In *Proc. of the 8th Int. Conf. on Educ. Data Mining*, Madrid, Spain, 2015.

[8] F. Harrak, F. Bouchet, V. Luengo, and R. Bachelet. Automatic Identification of Questions in MOOC Forums and Association with Self-Regulated Learning. In *Proc. of the 12th Int. Conf. on Educ. Data Mining*, pages 564–567, Montréal, Canada, July 2019.

[9] E. Huang, H. Valdiviejas, and N. Bosch. I'm Sure! Automatic Detection of Metacognition in Online Course Discussion Forums. In *Proc. of the 8th Int. Conf. on Affective Computing and Intelligent Interaction (ACII)*, pages 1–7, Sept. 2019.

[10] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. *Advances in Neural Information Processing Systems*, 30, 2017.

[11] S. M. Lundberg and S.-I. Lee. A Unified Approach to Interpreting Model Predictions. *Advances in Neural Information Processing Systems*, 30, 2017.

[12] I. Molenaar, A. Horvers, R. Dijkstra, and R. S. Baker. Personalized visualizations to promote young learners' SRL: the learning path app. In *Proceedings of the Tenth International Conference on Learning Analytics & Knowledge*, pages 330–339, 2020.

[13] J. Park, R. Yu, F. Rodriguez, R. Baker, P. Smyth, and M. Warschauer. Understanding Student Procrastination via Mixture Models. In *Proc. of the 11th Int. Conf. of Educ. Data Mining*, pages 187–197, Buffalo, NY, USA, 2018.

[14] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin. CatBoost: unbiased boosting with categorical features. *Advances in Neural Information Processing Systems*, 31, 2018.

[15] J. L. Sabourin, L. R. Shores, B. W. Mott, and J. C. Lester. Understanding and Predicting Student Self-Regulated Learning Strategies in Game-Based Learning Environments. *Int. J. of Artificial Intelligence in Education*, 23(1):94–114, Nov. 2013.

[16] W. Schneider. The Development of Metacognitive Knowledge in Children and Adolescents: Major Trends and Implications for Education. *Mind, Brain, and Education*, 2(3):114–121, 2008.

[17] D. H. Schunk and B. J. Zimmerman. Self-Regulation and Learning. In *Handbook of Psychology, Second Edition*. American Cancer Society, 2012.

[18] J. R. Segedy, J. S. Kinnebrew, and G. Biswas. Using Coherence Analysis to Characterize Self-Regulated Learning Behaviours in Open-Ended Learning Environments. *Journal of Learning Analytics*, 2(1):13–48, May 2015.

[19] D. Stipek, S. Recchia, and S. McClintic. Self-evaluation in young children. *Monographs of the Society for Research in Child Development*, 57(1):1–98, 1992.

[20] M. Taub and R. Azevedo. Using Sequence Mining to Analyze Metacognitive Monitoring and Scientific Inquiry based on Levels of Efficiency and Emotions during Game-Based Learning. *Journal of Educ. Data Mining*, 10(3):1–26, Dec. 2018.

[21] P. H. Winne and R. S. J. d. Baker. The Potentials of Educ. Data Mining for Researching Metacognition, Motivation and Self-Regulated Learning. *Journal of Educ. Data Mining*, 5(1):1–8, May 2013.

[22] B. J. Zimmerman. Investigating Self-Regulation and Motivation: Historical Background, Methodological Developments, and Future Prospects. *American Educational Research Journal*, 45(1):166–183, Mar. 2008.