A Longitudinal Study on Student Persistence in Programming Self-assessments*

Cheng-Yu Chung
Arizona State University
Tempe, Arizona
Cheng.Yu.Chung@asu.edu

Yancy Vance Paredes
Arizona State University
Tempe, Arizona
yvmparedes@asu.edu

Mohammed Alzaid
Arizona State University
Tempe, Arizona
Mohalzaid@asu.edu

Kushal Reddy Papakannu
Arizona State University
Tempe, Arizona
kushalreddy95@gmail.com

I-Han Hsiao
Arizona State University
Tempe, Arizona
Sharon.Hsiao@asu.edu

ABSTRACT

Self-assessment is an educational practice that helps students evaluate their own learning by distributed practices. This evaluation potentially has an effect on student's self-efficacy and therefore can influence their choice of activities and the likelihood of their success. The variation in the self-assessing behavior of students over the course of learning is often less explored. For instance, a student's short-term behavior may not necessarily infer how they will behave in the long-term. It is unclear how such development in their self-assessing behavior is related to academic performance and the corresponding self-assessment strategies. This longitudinal study aims to fill the gap by examining a self-assessment platform used in an introductory programming class from three different semesters. We analyzed the activity logs and modeled students' short-term and long-term study persistence on the platform using a probabilistic mixture model. The results suggest that short-term persistence was not related to short-term performance. However, the performance in the final exam was associated with earlier persistence patterns. A further analysis showed that low-performing students who maintained the self-assessment pattern improved in exams. Nevertheless, this longitudinal study contributes empirical evidence to the understanding of the development of self-assessment behavior in relation to academic performance.

Keywords

self-assessment, self efficacy, study persistence, programming concepts, learning analytics, computing education, probabilistic mixture model

1. INTRODUCTION

Engagement and persistence in learning have been considered as a key to attaining achievements in computing education [19]. Researchers in self-regulated learning (SRL) have shown that there is a relationship between students' belief in the effectiveness of learning strategies and such motivational responses [36]. This belief about one's "perceived capabilities for learning" is referred to as *self-efficacy* [31]. This not only affects how a student assesses his or her own learning outcome at the moment but also how the student chooses certain tasks and adapt to particular learning strategies.

Self-assessment is an educational practice that helps students evaluate their learning condition [7]. This practice can be extended by the theory of spacing effect and distributed practices for the provision of continuous evaluation of learning outcomes that can help students improve in a course [3]. When students keep receiving learning feedback from such a tool and the outcomes are attributed to the effort in self-assessment [21], their belief in self-efficacy may change [30] and therefore may be able to adjust and adapt their learning strategies to fit the best their conditions [2]. Research has shown that such temporally spaced and distributed practices are better than "compressed" ones in memory research [6]. There have been also research trying to correlate students' activity traces of self-assessments to SRL [15].

Following the train of thought about the relationship between self-assessment, self-efficacy, and learning strategies, we further hypothesize that students' self-assessment behavior is not stationary throughout the course of learning [14]. It is intuitive to have an impression that active and higher usage of self-assessment should be positively correlated to student's performance. However, in our previous work we observed that it was not always the case in our subjects. We found that students made adjustment to the usage of self-assessment according to, hypothetically, the attribution of effectiveness in terms of exam performance [14]. An active user did not necessarily end up with a higher performance in the exam. Moreover, there has been research examining the effectiveness of self-assessment in terms of memory, long-term retention, and cognitive outcomes like motivation, persistence, and self-efficacy, however, there are only few research papers focused on the explanation of variance between the changes of self-assessment behavior and students' performance in a course. Therefore, this work aims to examine the dynamics of the self-assessment usage pattern, which is referred to as *persistence pattern* onward, and evaluate how it is correlated to the variance in the performance in exams. Specifically, this work is guided by the following research questions:

RQ1 What are the persistence patterns of self-assessments in students from an introductory computer programming course? Are these patterns generalizable for students from different semesters?

RQ2 What is the relationship between the dynamics of persistence patterns and the variance in exam performance in the course? What are the changes that positively or negatively correlated to the performance?

RQ3 What are the effective practices of self-assessment for students whose performance is relatively low in the course? What are such practices for students who have relatively higher performance?

We have organized the rest of this paper in the following way. In the next section, we discussed related works in SRL in computing education and behavioral analytics in programming learning. Section 3 describes how we modeled the persistence patterns using a probabilistic model. Section 4 illustrates our findings of the dynamics of persistence pattern in relation to the variance in students' exam performance, which is followed by a discussion of the result with previous work and SRL theory. Finally, the conclusion and limitations of the model are described in Section 6.

2. RELATED WORK

2.1 Self-regulated Learning and Academic Success in Computing Education

Theories and practices of SRL have been established and evaluated since the late 1990s with the focus on student development, cognitive-behavioral processes, and social and motivational aspects [36]. On the line of social and motivational aspects, researchers discussed the construct *feedback cycle* within which a student experiences choosing a task of interest [5], judging the performance and comparing it to a standard [5, 36], building a perception of self-efficacy and being persistent on the process [29]. To assess multifaceted SRL behavior, various methodological instruments can be employed for different constructs, e.g., diary for personal and offline events, SRL scales for self-efficacy, "online" think-aloud protocols for SRL processes occurring during the learning [36]. Discipline-specific strategies of SRL in computing education has also been examined in the context of programming problem-solving [22], self-awareness [23], and metacognitive strategies [16].

In recent years, researchers from educational data mining (EDM) have started the discussion of applying EDM methods, which have a focus on using computerized methodologies for linking students' trace records to performance metrics by which researchers can optimize the learning process on both research and practice side. For example, Winne and Backer discussed such a potential in terms of SRL: "Self-regulated learning is a behavioral expression of metacognitively guided motivation" [34]. Students' activity and trace records could be a source of information about the process of learning, in which the challenge is to "obtain representations of learning as it unfolds...that are clearly and precisely matched to what theory describes." [34]. An example of such a theory is the 4-phase model of SRL proposed by Winne and Hadwin: 1) identification of resources, 2) goal setting, 3) carrying out the task, 4) reviewing the work [35, 34]. In this model, learners are assumed to be agents who decide and chose what to do depending on information from the environment, e.g., feedback, assessment outcomes, etc. Such an SRL procedure may not follow a certain sequence of phases ("weakly sequenced") and the results from it can be used recursively for the current of successive SRL phases [34].

The subject of this work is the relationship between student's self-assessment behavior, changes of such behavior, and how they are correlated with the performance in a course. We hypothesize that this process can be considered an "unfolding" process following the 4-phase model of SRL from Winne and Hadwin where students, who use the system based on their own decision, try to obtain some feedback (resources) on their understanding of learning topics and therefore can identify what they need to further study on.

2.2 Behavioral Analytics in Programming Learning

Behavioral analytics is an area of research that is gaining popularity. It traces its roots from research on understanding data captured from e-commerce. Research in this field, such as exploratory studies, was driven by the advancement of technology where systems became capable of capturing large quantities of data which may come from multiple sources. There has been growing interest in exploring the application of behavioral analytics to education data to support pedagogy. This spans from student's performance prediction, intelligent course recommendation, data-driven learning analytics, and personalized learning [12]. Some of these education data are considered to be ambient data (accretion data) that learners generate [33] while using learning environments. This could be in the form of capturing the event where a user clicks on a hyperlink to open a web resource. That can capture the user's cognition and motivation. In another work, sequential analysis was applied to behavioral data to explore how it was affected by the motivation of students to learn [32]. They indicated that online reading duration in the online learning system was a better indicator of reading seriousness in learners. Another work proposes to extend how behavioral analytics is perceived [4]. In this case, they proposed to look into investigating the deviation of a student from a normal behavior. This normal behavior is contextually dependent on the issue at focus.

Modeling student's learning is an ongoing research in the field. Such student models reside in intelligent tutoring systems or any adaptive educational systems. In these systems, behavior logs are often used to estimate students' learning (i.e., interaction with the tutors which results in updates on the knowledge components). In the context of learning a programming language, several parameters have been used to estimate the coding knowledge of students. This

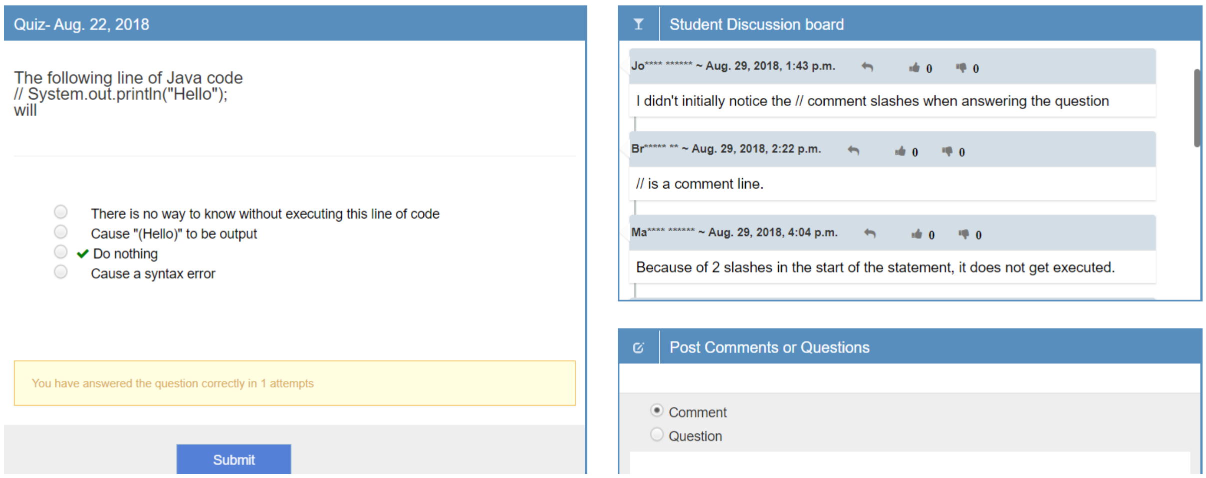

Figure 1: A Screenshot of the Self-assessment Platform. The left panel shows the question and options. The right panel shows the feedback/discussion from other students (which is only accessible after the student answers a question).

includes the sequence of success in programming problem-solving [17]; how the students progressed in solving programming assignments [26]; the dialogic strategies between students [8]; identifying the strategies of students when seeking information related to programming [24]; assignment submission compilation behavior [1, 20]; how students troubleshoot and test their solutions [9]; and code snapshot process state [11].

3. METHODOLOGY

3.1 Research Platform

The research platform [3] utilized in this study is a homegrown system designed as an educational tool grounded in learning science principles (Figure 1). It is based on distributed practice [3], retrieval practice and testing effects [27], reflection and metacognition [18], feedback [10], peer interaction [28]

This platform acts as a supplemental self-assessment tool for introductory programming courses. It provides students with small distributed opportunities to master their programming knowledge. It publishes daily questions to measure the learning of a specified programming knowledge component and provide extended learning and reflecting opportunities to the students. The design rationale of the system is based on the following learning concepts:

- Distributed practices: rather than having the content presented at once, the learning becomes more effective when broken into chunks. This strategy is even more effective with constant increments of small practices over time.
- Retrieval practice and testing effects ensure that the student remembers what they have learned. It also enhances long-term retention.
- Reflection and metacognition encourage students to take the time and think about the learned content and what the student thinks about it that helped him or her to develop and grow.
- Feedback: When the student receives feedback, it facilitates their development as independent learners. When the feedback is immediate it helps the students evaluate and regulate their learning at their own pace.
- Peer interaction: The benefit of peer interaction in learning is significant, Therefore, a designated discussion board for each question was provided to facilitate this interaction. This also increases the social benefit from the reflections.
- Persistent and regularity: Providing one multiple-choice question a day keeps the student interested to check for newly posted questions and encourage them to practice regularly.

Semester	# of Students	Statistics of Attempts
Fall 2016	217	$M = 26.90, SD = 29.23$
Spring 2018	112	$M = 17.59, SD = 27.20$
Fall 2018	211	$M = 9.50, SD = 21.46$

Table 1: Statistics of Datasets

3.2 Data Collection

As students access the system, they are prompted with the quiz of the day. They can attempt the question right away or leave it for later and move to the question history. The system allows the students to attempt a question multiple times until the correct answer is selected. Each attempt is marked with the appropriate flag indicating the review source (quiz of the day, review, attempt & retry) and whether the student answered correctly or not. At the beginning of the course, students were encouraged to reflect on their attempts. In the system, they are prompted to reflect right after an attempt through the discussion board where they can interact with peers. The credentials of the peers are anonymized to preserve privacy and to facilitate unbiased discussions and interactions. Finally, students can access previously posted questions at any time using the calendar feature or the question history list.

In this study, we collected data from an introductory computer programming course offered in a university. The dataset was from three different semesters. An overview of the raw dataset is shown in Table 1. After dropping those students without grade or any activity on the platform, we were left with 344 students for the analysis (the number of attempts: $M = 17.68, SD = 28.25$).

3.3 Discovering Persistence Patterns by Probabilistic Mixture Model

Activity stream data is known for its rich properties such as analyzing students' time management behaviors [25]. The activity made by a student during practice such as submitting an answer to a question is recorded as transactions in the activity-stream data. To determine the persistence, we consolidated the click-stream data by counting the number of times transactions were recorded in each week [14, 13]. The data was then grouped into three exam periods. A mixture model was applied to the activity stream. Our exploratory analysis revealed three micro patterns: *Active*, *Cramming*, and *Inactive*. The Active represents the students who practice actively in all the weeks in a given time-frame. The Cramming represents the students who use the platform only right before an exam. The Inactive represents users who use the platform minimally.

To ground the identification of these patterns, we adapted tools from time-series analysis. A moving average model (MA) with a time-lag of 1 was applied to the component averages of patterns. Each exam period where more than half of values were less than 0.05 (the minimum value of components after normalization) was marked as *Condition 1*. To capture the peak of changes in the amount of activity, we also calculated the difference and marked exam periods with the value beyond $M+SD$ or $M-SD$ as *Condition 2*. Afterward, the tagging of patterns was done in this order: if an exam period is marked as Condition 2, tag it as *Cramming*; if it is marked as Condition 1, tag it as *Active*; otherwise, tag it as *Inactive*. An illustration of these characteristics and the tagging process is shown in Figure 2.

4. RESULTS

The major goal of this study is to find out the correlation between students' persistence patterns and their performance in the class. A *macro* (long-term) persistence pattern consists of three *micro* (short-term) patterns distributed in the three exam periods. The assumption behind this model is that a student's effort can be represented by a sequence of events where a later event (e.g., an exam score) or an effort (e.g., the decision to study every week actively) is the decision based on the past events. Specifically, a student's activity is modeled by a sequence $(P1, E1, P2, E2, P3, E3)$ where $P1, P2, P3$ are three micro persistence patterns and $E1, E2, E3$ are the normalized performance of three exams in the course. The composition of $P1, P2, P3$ is referred to as macro persistence pattern.

Out of records from 344 students, we found 10 different macro persistence patterns. The majority of students were categorized into the macro pattern *CCI* ($\sim$ 44%) and *CII* ($\sim$ 35%). The distribution of found patterns is shown in Table 2. This result is not surprising because the use of the platform every week was not mandatory and students tended to study intensively right before the exam regardless of the reason.

Pattern	Quantity	Ratio
CCI	151	0.44
CII	119	0.35
ACC	24	0.07
CCC	14	0.04
AAC	10	0.03
ACI	8	0.02
ICC	6	0.02
ICI	5	0.01
ACA	4	0.01
IIC	3	0.01

Table 2: The Distribution of Persistence Patterns

4.1 Exploring the Relationship of Persistence Patterns and Exam Performance

Programming concepts are often complex and coupled. In a course about introduction to programming, we can expect that an advanced concept is usually built up from sets of fundamental concepts. The content of a later exam is inevitably accumulated from previous exams. To illustrate the complexity of accumulated programming concepts overtime, we calculated the correlation between exam performance and found out that the correlation between E1 and E2 (Pearson's $r(E1, E2)$) is 0.74; $r(E1, E3) = 0.71$; and $r(E2, E3) = 0.79$. We also found the partial correlation between E2 and E3 with controlled E1 is 0.56. Following the heuristic interpretation of Pearson's r, this result suggested that the performance of E1, E2, and E3 was moderately to highly correlated and we should consider the effect from previous exam performance when analyzing the variance in students' persistence patterns and exam performance.

Based on this result, we believed that the only period in a semester where we could observe the marginal correlation between persistence patterns and exam performance was the first exam period where P1 and E1 occurred. We hypothesized that different micro persistence patterns were related to the exam performance considering that students might improve their understanding of learning topics by active self-assessment on the platform. To test this hypothesis, we conducted a one-way ANOVA analysis on P1 and E1. The result showed that P1 did not have significant main effect on E1 ($F(2, 341) = 0.72, p = 0.48$), namely, micro/short-term persistence pattern was not marginally related to short-term performance.

In spite of the rejected hypothesis, we still observed that students with different P1 did not develop in the same way in the later exams (See Figure 3). We believed this probably indicated that in our sample group *students' short-term persistence had an effect on long-term development throughout the semester*. We tested this hypothesis by an ANOVA analysis of 3-by-3 factorial design over P1 and P2 on E2, which

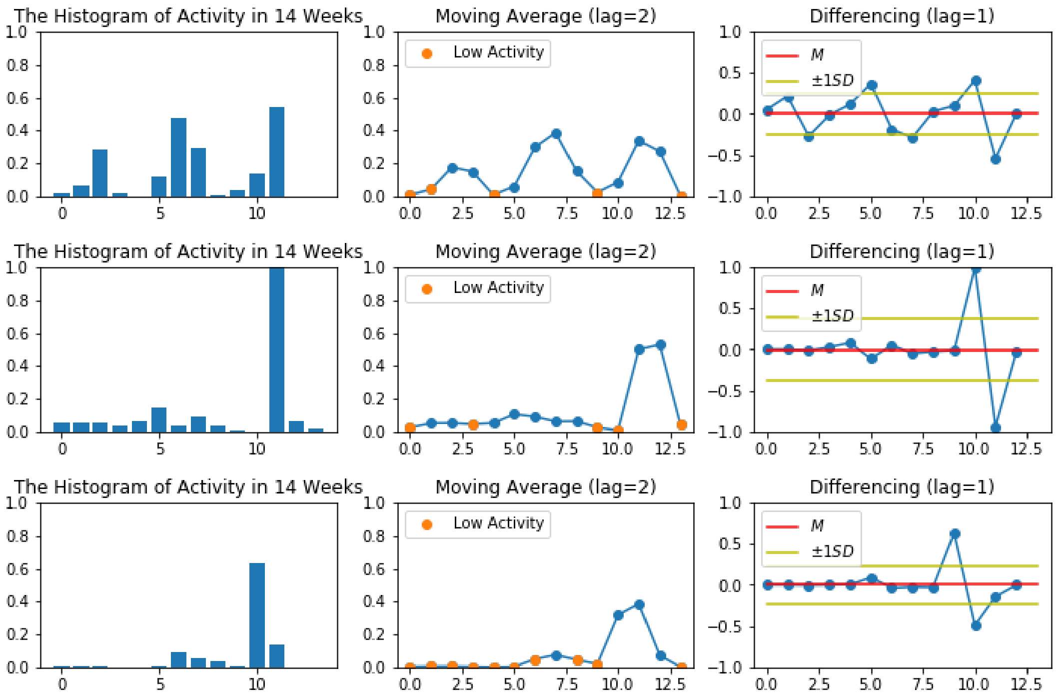

Figure 2: An Illustration of the Characteristics of Persistence Patterns. Each row represents the characteristics of a persistence pattern. The annotation of persistence pattern was done by checking whether the moving average and differencing passed predefined thresholds (see Section 3.3 for detail). From top to bottom, the annotated patterns are *CCC*, *AAC*, and *IIC*.

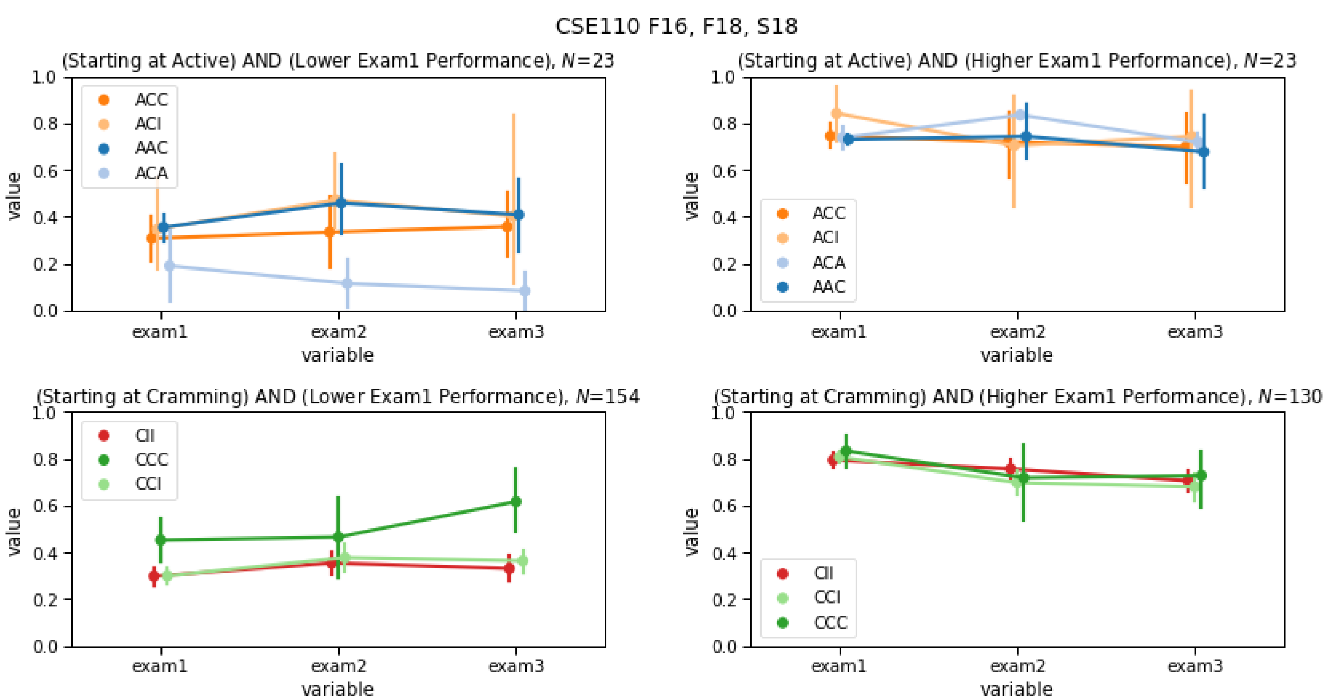

Figure 3: The Relationship of Exam Performance and Persistence Patterns for Students Starting with Active (A; the upper row) and Cramming (C; the lower row). The left column represents low-performing (LP) students and the right one represents high-performing (HP) students. The patterns of interest with significant difference were found mainly in the LP group, including the pairs (AAC, ACA), (CCC, CCI), (CCC, CII).

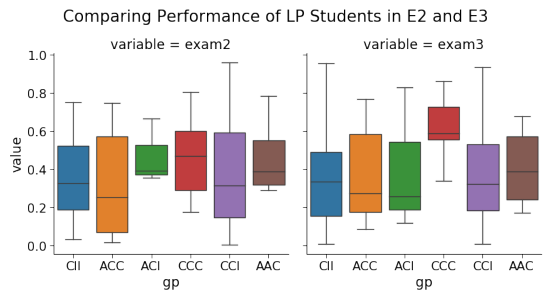

Figure 4: The Comparison Exam Performance in LP Students with Different Persistence Patterns. We can see that the group "CCC" seemed to outperformed all the other groups in E3.

did not show any significant main effects from P1, P2, or the interaction of P1 and P2. However, the test of factorial design over P1 and the interaction of exam pairs (i.e., the formula $P1 + P1 : P2 + P2 : P3 + P1 : P3$) on E3 showed that the main effect of the interaction of P1 and P2 was significant ($F(4, 327) = 3.25, p = 0.01$). Together with our previous test, this result suggested that even though short-term behavior might not bring an effect to the immediate exam performance, in the long run, a student's earlier behavior (P1 and P2) might have an effect on the performance of the final exam (E3).

4.2 Correlating the Dynamics of Persistence Patterns to Exam Performance

Our next question was about the relationship between the trajectory of persistence patterns (i.e., the variance in P1, P2, P3) and student performance throughout the course of three exams. We first grouped students into high-performing (HP) and low-performing (LP) by checking whether their performance in E1 was higher than 0.6 or not. This cut point was in accordance with the fact that 60% is a common cut point which decides the passing grade. The choice of E1 was based on the finding in Section 4.1. Moreover, this transformation made the analysis of multiple categorical groups cleaner and easier to follow. An overview of students' macro patterns and exam performance is shown in Figure 3.

4.2.1 Effective Persistence Patterns in LP Students Starting with Active

The Active micro pattern (A) represents a continuous effort on the self-assessment platform for a period of time (see Section 3.3 for the definition). In the sample dataset, we identified four macro patterns starting with A: ACC, ACI, AAC, and ACA. Among these patterns, we found that the development of AAC and ACA students was of interest. First, LP students with AAC (LP-AAC) had similar performance in E1 and E2 compared to LP-ACA students. However, LP-AAC students ($M = 0.41, SD = 0.19$) performed significantly better than LP-ACA ($M = 0.08, SD = 0.08$) ($t(6) = 2.82, p = 0.05, d = 1.659$). One apparent difference in these two groups of students was that LP-ACA changed from A to C in the second period, and changed back to A in the third period. On the other hand, the persistence pattern of LP-AAC was relatively consistent in the first two periods and changed to Cramming in the third period.

This result indicated that keeping Active in the first two periods might be helpful for LP students. This was probably due to the difficulty of learning topics in the course. Since topics in E1 and E2 were important for students to build up the fundamentals of programming languages, if they did not self-assess actively, they might not know that they needed to catch up as soon as possible. Note the sample size of these pattern was extremely small. A future study with a large sample size is needed to cross-validate this finding.

4.2.2 Effective Persistence Patterns in LP Students Starting with Cramming

The Cramming micro pattern (C) represents an intense amount of effort on the self-assessment platform in a short period of time. This is a common pattern we can find when the time is close to an exam date. In our sample dataset, we found that the pairs (LP-CCC, LP-CCI), (LP-CCC, LP-CII), and (LP-CCC, HP-CCC) revealed interesting patterns in terms of the variance in exam performance.

For the first pair, (LP-CCC, LP-CCI), the analysis showed there was no significant difference in their performance of E2, however, in E3 LP-CCC ($M = 0.61, SD = 0.17$) performed significantly better than LP-CCI ($M = 0.33, SD = 0.21$) ($t(91) = 3.19, p = 0.02, d = 1.04$). When comparing LP-CCC to LP-CII ($M = 0.33, SD = 0.21$), the significance was only found, again, in E3 ($t(65) = 3.60, p = 0.01, d = 1.36$). In other words, LP students starting with Cramming and keeping this persistence pattern across the semester performed better than those who did not keep the persistence pattern (CCI and CII) in the final exam. We also found that LP-CCC students even had the best performance in E3 compared to those with other macro patterns ($M = 0.35, SD = 0.23$) (see Figure 4; $t(175) = 3.48, p = 0.02, d = 1.15$). An ANOVA analysis of 3-by-3 factorial design over P2*P3 on E3 also showed that the main effect from P3 was significant ($F(1, 144) = 6.73, p = 0.01$), which further emphasized the importance of the persistence pattern in the final period. Moreover, when comparing LP-CCC students to their high performing sibling, HP-CCC, we found that although in E1 LP-CCC ($M = 0.45, SD = 0.12$) performed worse than HP-CCC ($M = 0.83, SD = 0.10$) ($t(12) = -5.80, p = 0.00, d = -3.23$; which was mainly due to the grouping), their performance in E2 and E3 was not significantly different.

The analysis showed that LP-CCC students not only outperformed those with other macro persistence patterns C in the LP group, but performed on a par with HP students with the same pattern in the final exam. These results collaboratively suggested that being consistent on the Cramming behavior was an effective practice for LP students. One possible explanation was that LP students who kept the Cramming behavior on the self-assessment platform throughout the semester might be showing their grip and willingness to improve in the class. Another possible assumption of this effective practice was that intensive self-assessment helped students to identify the learning topics or concepts they need to further review and study.

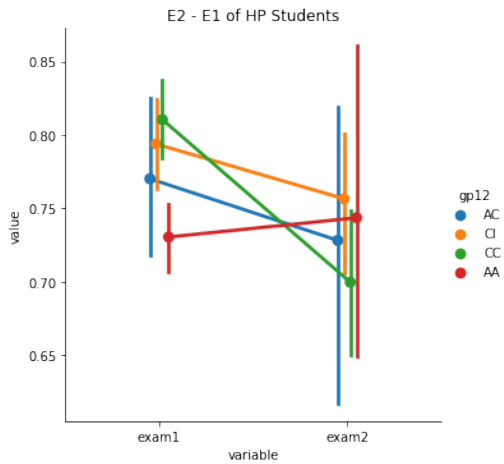

Figure 5: Comparing Exam Performance of High Performing Students in Exam1 and Exam2. We can see that the performance of students with the pattern "CC" dropped the most compared to students with the other patterns. Only students with the pattern "AA" had the tendency to improve the performance.

4.2.3 Effective Persistence Patterns in HP Students

The performance of HP students was relatively stable compared to LP students (Figure 3). An ANOVA analysis of 3-by-3 factorial design over P1 and P2 on the difference of E1 and E2 showed that the main effect of the interaction of P1 and P2 was significant ($F(2, 148) = 3.40, p = 0.03$). Following this outcome, we further compared and examined the value with different persistence patterns (see Figure 5). The only pattern of interest we found was that HP students starting with Cramming and kept doing so were not able to keep their performance in E2. A statistics test showed that the performance of this group of students dropped significantly from E1 ($M = 0.81, SD = 0.11$) to E2 ($M = 0.70, SD = 0.21$) ($t(142) = -3.85, p = 0.00, d = -0.65$). Such a pattern was not found from E2 to E3.

This result could be a signal that for HP students who wanted to stay competitive, intensive self-assessment might not help much. A possible explanation to this outcome was that due to that the topics on the self-assessment platform were "limited" in terms of scope and the amount of content, when the complexity of topics increased, students were not able to use the platform to review the important and necessary learning content that were not covered by the self-assessment platform.

5. DISCUSSION

One SRL strategy in the literature relevant to the performance of self-assessment is *assessment of task difficulty* proposed by Falkner and colleagues. When the identification of needed skills is incorporated, it would lead to the development of time management and sub-goal planning [16]. The performance of self-assessment has two fundamental goals: for students to practice learning concepts, and for them to obtain feedback about their current span of knowledge. While the former is explicit, the latter is rather implicit from the perspective of the students. It is assumed that as students use the self-assessment platform, they become aware of which learning content they currently need to improve on ("the identification of needed skills"). This will allow them to review and address these learning gaps accordingly ("the development of time management and subgoal planning"). Following this hypothesis, the practice of self-assessment may reflect a part of SRL strategies which allows for the interpretation of the findings in this work in terms of SRL behaviors.

Our analysis showed that low-performing students who kept the cramming persistence pattern in self-assessment were able to improve and achieve competitive performance with the high-performing counterpart in the final exam. This result suggests that although generally, cramming or procrastination is a less-desirable behavior in learning [25], when such behavior is found in self-assessment platforms, a positive outcome may be seen. An optimistic explanation to this result is that the platform was seen as supplemental material to the course since it follows a format that closely resembles the formal assessments (i.e., multiple-choice questions). Thus, students who wanted to improve their performance could obtain *actionable* feedback allowing them to review exam content in a short period of time. On the other hand, a relatively pessimistic explanation is that students were simply gaming the system to memorize the content in the hopes of seeing similar questions in the exams.

One interesting pattern we found from the analysis was that the effect of early behavior might reflect on performance at a later time (see Section 4.1). Our analysis showed that low-performing students who kept the same persistence patterns in the first two exam periods might perform better than the others. This result may suggest that 1) in this course, the effort in the first two exam periods was crucial, which is not surprising considering the comprehensive nature of midterm exams; and 2) being persistent only for a short term was not enough. This can be leveraged to guide students in practice. For example, a recommender can be implemented to inform students that being persistent in the long run is important. This recommender can also adapt to students' self-assessments in a short term.

This study gave us a glimpse of the students' behavior as they progressed into the course. We observed how many students utilized the system and crammed as they prepared for an upcoming exam. Those who kept their persistence patterns outperformed those who did not in terms of performance in the formal assessments. This result suggested that persistently putting an effort to utilize additional materials, in general, would positively be reflected on the course performance. However, we did not look into their self-reported motivations or reasons for coming back to the system, e.g., whether it was for them to self-assess or to practice on to additional learning materials. Since what we know about the students is currently limited to their activity log data on the system, we may only assume that those who persistently used the system belonged to the so-called "hard-working" ones who had been the better-performing students prior to taking the course or had a better metacognitive skills.

Additionally, although the findings may have a potential connection to SRL, however, in this work we are not able to ground this hypothesis due to the lack of data from authentic SRL measurements (e.g., qualitative questionnaires for mapping constructs which are widely used in literature). Additionally, the lack of some persistence patterns or sufficient samples potentially makes our interpretation biased toward a certain kind of persistence patterns. This is despite the fact that the analyses were based on data collected from the same course given in three different semesters, which to some extent had consolidated the possible patterns that could be found in this specific course. These are considered the current limitations of this work and a future study can further evaluate our findings in consideration of these issues.

6. CONCLUSIONS AND LIMITATIONS

This work examined the relationship between the persistence of self-assessment and the exam performance in an introductory course of computer programming. We collected data from the same course given in three different semesters. Out of 344 students, we identified 10 different long-term persistence patterns by a probabilistic mixture model, each of which consisted of three short-term persistence patterns including Active, Cramming, and Inactive. From a series of analyses which explained the variance of exam performance by the dynamics or changes in persistence patterns, we found that low-performing students benefited from the continuity of intensive or active self-assessment; and, somewhat, on the contrary, high-performing students might not achieve similar effectiveness by the continuity of cramming. We discussed these outcomes under the framework of self-regulated learning and provided possible assumptions and explanations in the context of the course.

There are some limitations to the methodology of this work. First, the persistence model in use was built on the amount of students' activities on the self-assessment platform which only included the count of unique question attempts. In other words, our model did not consider other probably valuable information in self-assessment such as the correctness of first attempts, the coverage or difficulty of learning topics, among others. The second limitation is that our definition of the micro persistence pattern Cramming may overlap Active and Inactive to some extent. Specifically, we did not discriminate students who were "Active and Cramming" from those who were "Inactive and Cramming". One reason for this decision was to avoid the overfitting that made the result of patterns too sparse. Nevertheless, we might introduce some bias toward the Cramming pattern in our analysis. Finally, the persistence patterns found in the same course seemed consistent or similar in the three different semesters. However, this could be due to the structure of the content or the nature of content provided in this specific course of computer programming. Although we can expect that some behavioral patterns may be general across the field of study (e.g., cramming before the exam), our findings related to the exam performance may not be the case. Future research should take the organization of course content into account when trying to replicate the result in a different course.

7. REFERENCES

[1] A. Altadmri and N. C. Brown. 37 million compilations: Investigating novice programming mistakes in large-scale student data. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*, pages 522–527. ACM, 2015.

[2] M. Alzaid and I.-H. Hsiao. Behavioral Analytics for Distributed Practices in Programming Problem-Solving. In *2019 IEEE Frontiers in Education Conference (FIE)*, volume 2019-Octob, pages 1–8. IEEE, 10 2019.

[3] M. Alzaid, D. Trivedi, and I.-H. Hsiao. The effects of bite-size distributed practices for programming novices. In *2017 IEEE Frontiers in Education Conference (FIE)*, volume 2017-Octob, pages 1–9. IEEE, 10 2017.

[4] A. R. Baig and H. Jabeen. Big data analytics for behavior monitoring of students. *Procedia Computer Science*, 82:43–48, 2016.

[5] A. Bandura and D. H. Schunk. Cultivating competence, self-efficacy, and intrinsic interest through proximal self-motivation. *Journal of Personality and Social Psychology*, 1981.

[6] A. S. Benjamin and J. Tullis. What makes distributed practice effective? *Cognitive Psychology*, 2010.

[7] D. Boud and N. Falchikov. Quantitative Studies of Student Self-Assessment in Higher Education: A Critical Analysis of Findings. *Higher Education*, 18(5):529–549, 1989.

[8] K. E. Boyer, R. Phillips, A. Ingram, E. Y. Ha, M. Wallis, M. Vouk, and J. Lester. Investigating the relationship between dialogue structure and tutoring effectiveness: a hidden markov modeling approach. *International Journal of Artificial Intelligence in Education*, 21(1-2):65–81, 2011.

[9] K. Buffardi and S. H. Edwards. Effective and ineffective software testing behaviors by novice programmers. In *Proceedings of the ninth annual international ACM conference on International computing education research*, pages 83–90. ACM, 2013.

[10] D. L. Butler and P. H. Winne. Feedback and Self-Regulated Learning: A Theoretical Synthesis. *Review of Educational Research*, 65(3):245–281, 9 1995.

[11] A. S. Carter, C. D. Hundhausen, and O. Adesope. The normalized programming state model: Predicting student performance in computing courses based on programming behavior. In *Proceedings of the eleventh annual International Conference on International Computing Education Research*, pages 141–150. ACM, 2015.

[12] L. Cen, D. Ruta, and J. Ng. Big education: Opportunities for big data analytics. In *2015 IEEE International Conference on Digital Signal Processing (DSP)*, pages 502–506. IEEE, 2015.

[13] C.-Y. Chung and I.-H. Hsiao. Quantitative Analytics in Exploring Self-Regulated Learning Behaviors: Effects of Study Persistence and Regularity. In *2019 IEEE Frontiers in Education Conference (FIE)*, pages 1–9. IEEE, 10 2019.

[14] C.-Y. Chung and I.-H. Hsiao. Investigating Patterns of Study Persistence on Self-Assessment Platform of Programming Problem-Solving. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, pages 162–168, New York, NY, USA, 2

2020. ACM.

[15] A. Cicchinelli, E. Veas, A. Pardo, V. Pammer-Schindler, A. Fessl, C. Barreiros, and S. Lindstädt. Finding traces of self-regulated learning in activity streams. *Proceedings of the 8th International Conference on Learning Analytics and Knowledge - LAK '18*, pages 191–200, 2018.

[16] K. Falkner, R. Vivian, and N. J. Falkner. Identifying computer science self-regulated learning strategies. In *Proceedings of the 2014 conference on Innovation & technology in computer science education - ITiCSE '14*, pages 291–296, New York, New York, USA, 2014. ACM Press.

[17] J. Guerra, S. Sahebi, P. Brusilovsky, and Y. Lin. The Problem Solving Genome: Analyzing Sequential Patterns of Student Work with Parameterized Exercises. In *7th International Conference on Educational Data Moining*, pages 153–160, London, UK, 2014. International Educational Data Mining Society.

[18] D. J. Hacker, J. Dunlosky, and A. C. Graesser. *Metacognition in Educational Theory and Practice*. Routledge, 3 1998.

[19] B. Hoffman, R. Morelli, and J. Rosato. Student Engagement is Key to Broadening Participation in CS. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education - SIGCSE '19*, pages 1123–1129, New York, New York, USA, 2019. ACM Press.

[20] M. C. Jadud and B. Dorn. Aggregate compilation behavior: Findings and implications from 27,698 users. In *Proceedings of the eleventh annual International Conference on International Computing Education Research*, pages 131–139. ACM, 2015.

[21] H. H. Kelley and J. L. Michela. Attribution Theory and Research. *Annual Review of Psychology*, 1980.

[22] D. Loksa and A. J. Ko. The Role of Self-Regulation in Programming Problem Solving Process and Success. In *Proceedings of the 2016 ACM Conference on International Computing Education Research*, pages 83–91, New York, NY, USA, 8 2016. ACM.

[23] D. Loksa, A. J. Ko, W. Jernigan, A. Oleson, C. J. Mendez, and M. M. Burnett. Programming, Problem Solving, and Self-Awareness. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 1449–1461, New York, NY, USA, 5 2016. ACM.

[24] Y. Lu and I.-H. Hsiao. Seeking programming-related information from large scaled discussion forums, help or harm?. In *Proceedings of the 9th International Conference on Educational Data Mining*, pages 442–447, 2016.

[25] J. Park, R. Yu, F. Rodriguez, R. Baker, P. Smyth, and M. Warschauer. Understanding Student Procrastination via Mixture Models. In *Educational Data Mining*, pages 187–197, Buffalo, NY, 2018. International Educational Data Mining Society.

[26] C. Piech, M. Sahami, D. Koller, S. Cooper, and P. Blikstein. Modeling how students learn to program. In *Proceedings of the 43rd ACM technical symposium on Computer Science Education*, pages 153–160. ACM, 2012.

[27] H. L. Roediger and A. C. Butler. The critical role of retrieval practice in long-term retention. *Trends in Cognitive Sciences*, 15(1):20–27, 1 2011.

[28] R. D. Roscoe and M. T. H. Chi. Tutor learning: the role of explaining and responding to questions. *Instructional Science*, 36(4):321–350, 7 2008.

[29] D. H. Schunk. Enhancing self-efficacy and achievement through rewards and goals: Motivational and informational effects. *Journal of Educational Research*, 1984.

[30] D. H. Schunk. Sequential attributional feedback and children's achievement behaviors. *Journal of Educational Psychology*, 76(6):1159–1169, 1984.

[31] D. H. Schunk and E. L. Usher. Assessing self-efficacy for self-regulated learning. In B. J. Zimmerman and D. H. Schunk, editors, *Educational psychology handbook: handbook of self-regulation of learning and performance*. Routledge, 2011.

[32] J. C. Sun, C. Lin, and C. Chou. Applying learning analytics to explore the influence of online learners' motivation on their online learning behavioral patterns. In *2016 5th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI)*, pages 377–380, 2016.

[33] E. J. Webb, D. T. Campbell, R. D. Schwartz, and L. Sechrest. *Unobtrusive measures*, volume 2. Sage Publications, 1999.

[34] P. H. Winne and R. S. Baker. The Potentials of Educational Data Mining for Researching Metacognition, Motivation and Self-Regulated Learning. *JEDM - Journal of Educational Data Mining*, 2013.

[35] P. H. Winne and A. F. Hadwin. Studying as self-regulated learning. In *Metacognition in educational theory and practice*. Routledge, 1998.

[36] B. J. Zimmerman and D. H. Schunk. Self-Regulated Learning and Performance: An Introduction and an Overview. In *Educational psychology handbook: handbook of self-regulation of learning and performance*, page 1. Routledge, 2011.