Collecting and Integrating Multimodal Data from a Programming Exercise Environment

Yuta Taniguchi
Kyushu University
taniguchi@ait.kyushu-u.ac.jp

Atsushi Shimada
Kyushu University
atsushi@ait.kyushu-u.ac.jp

ABSTRACT: Learning programming languages is one of the tough problems for students. This paper proposes four types of learning activities found in programming exercise contexts. We discuss their usefulness and how to collect such activities as data. Furthermore, we show two examples of analysis based on the data.

Keywords: Programming exercise, learning activity, e-textbooks

1 INTRODUCTION

Computer programming has been an essential skill for engineers and computer scientists for the past decades. Generally speaking, programming education for novice students are considered hard. For example, in C programming course, it is tough for beginners to write even a compilable source code from scratch, and they tend to write a program that will result in compile errors. However, since compile error messages are not always straightforward and beginner students do not have enough experiences to interpret error messages, fixing such errors takes a long time. Although such learning experiences could lower their motivation, not all students ask their teachers or friends when they have a trouble. Therefore, it is inevitable for teachers to actively intervene in programming exercise.

From the perspective of learning analytics, temporal records of students' learning process give plenty of insights to decide who and when we should support. For example, (Blikstein, 2011) collected logs of students' actions and analyzed coding strategies and temporal change of code, (Fu, Shimada, Ogata, Taniguchi & Suehiro, 2017) presented a dashboard system that visualize current situations of students' progress in programming exercise in real-time fashion. However, most of the existing studies only focused on coding activities.

Students' ability to find information necessary for making progress on their exercise has been rarely considered. For example, if we want to know whether a student will need a help or not, evaluating such abilities would be important as well as the ability to solve errors. Thus, we need novel approaches that take into account the information acquisition processes such as referring external learning materials, and/or asking classmates/teachers. Collecting such activity logs and integrating them with exercise activity logs would enable us to know students' situations in detail and support them in much more effective way.

2 THE LEARNING ACTIVITY TYPES IN EXERCISE

Learning activities previously studied are limited. Considering where students acquire the necessary information or knowledge during programming exercise in face-to-face classrooms, we propose four learning activity types students do during exercise. As shown in Figure 1, there are four targets around a student with which a student possibly interacts as a part of exercise activities.

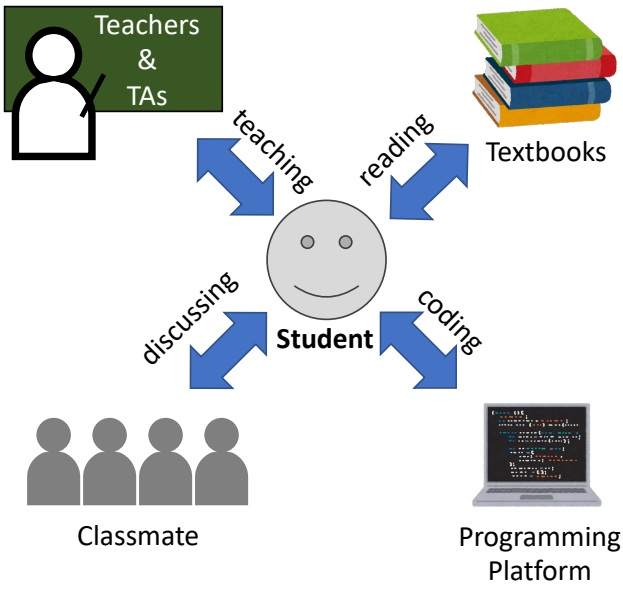

Figure 1. Four types of learning activities could be observed in programming exercise environments.

2.1 Interactions with Teachers and Teaching Assistants

Interacting with teachers and teaching assistants (TAs) are the representative activities found during exercise. Through lectures, teachers deliver important knowledge and information which would be necessary in exercise. Students usually follow a teacher and read the same page of a textbook as the teacher. In other situation, during exercise, for example, teachers and TAs would also directly interact with students to help them solve errors.

With e-textbook system, the former learning activity is relatively easy to record. For example, we can record which page a student is reading at a point of time. The latter type of interaction, however, needs additional mechanism since such interactions are usually done orally. The easiest way to collect such activities is asking teachers and TAs to record the names of students they interacted with. It might also be possible to use microphone for recording dialogue. Introducing a chat system for question answering would be another option that could be realized relatively easily.

2.2 Reading Textbooks

Students also read textbooks autonomously. For instance, during exercise, a student may consult a textbook for checking a grammar of a programming language to fix compilation errors. Even though

textbooks do not always provide information required to fix errors, they are definitely one of the authorities for students who do not have enough debug experiences. In addition, highlighting or note-taking are the other essential learning activities.

The usage of textbooks can be recorded with an e-book viewer system. Such a system is usually capable of storing users' activity logs into a database. Basically, page flipping actions and annotating actions can be collected. For more detailed reading behavior some researchers used eye trackers, but usually such devices cost so much that we cannot deploy at scale.

2.3 Coding Activities

As many literatures focused on, coding activity is a principal learning activity in programming exercise. Recording such activity tells us how a student worked on assignments. Monitoring students' activity in real-time give us chances for intervention to help students in trouble.

Usually a kind of educational programming platform is employed to record a student's actions, for example keyboard inputs, source cord, and compilation results.

2.4 Discussion with Classmates

Classmates also play an important role in exercise. Sometimes, experienced teachers or TAs cannot understand why a novice programmer does not understand concepts, and the difference of knowledge could be a barrier in the communication between teachers and students. Usually students have a relatively similar level of understanding, and it could be considered that the communication between students is much easier. Actually, we can easily find students having a talk with each other during exercise instead of asking teachers or TAs.

Since such discussions may include sensitive information and happen everywhere in a class, to collect this kind of activity logs would be difficult ethically and technically. One feasible solution could be to introduce online forums where every post is public and optionally participants are anonymized. This idea could have potential to reduce the risk of ethical problem and make it easy to collect data. However, there is still a problem that we need to have student actively use forums.

3 POSSIBLE ANALYSIS

In this paper, we focus on the modalities of reading textbooks and coding activities. Figure 2 shows an example of visualized learning processes consists of them for each student (a horizontal line), which is composed of reading events (yellow dots), successful compiling events (green dots), and failure compilation events (red dots). We can see diversity of exercise processes in the picture. The visualization would be helpful for teachers to understand the current situation of a class, and temporal analysis on the processes has potential to predict the adequate timing of intervention.

Figure 3 shows an example of analysis on such learning processes from our previous study (Taniguchi, Okubo, Shimada & Konomi, 2018). The picture shows the aggregated contributions of each textbook page to resolve each error. Pages are considered positively contributed if a student solved the error message after reading the page and so forth. This kind of analysis enable us to find the problem of

textbooks or to identify students who cannot solve compile errors which are poorly described in textbooks.

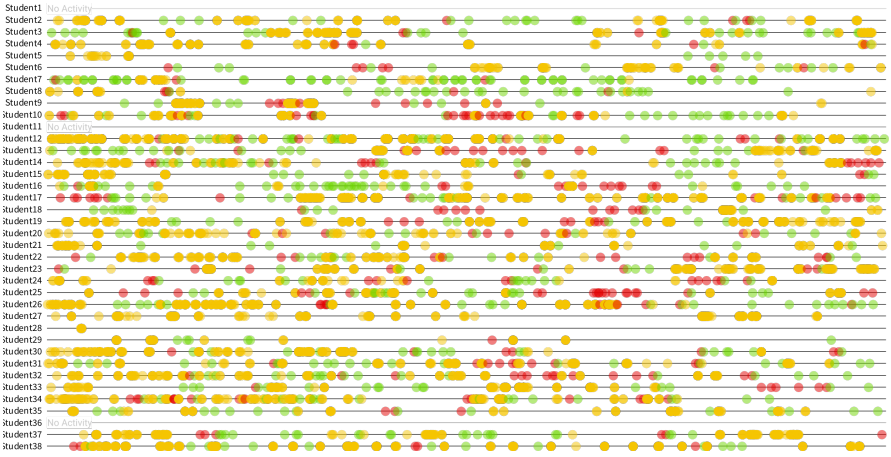

Figure 2. An example of learning processes of programming exercise consisting of e-book events and compilation events.

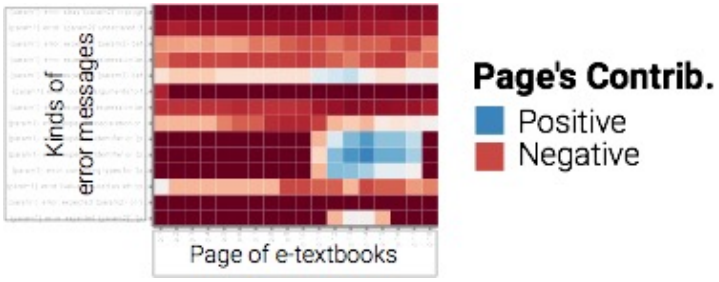

Figure 3. The result of page-error relation analysis. The positive or negative contribution of reading a specific page to solve a specific error is shown.

4 CONCLUSION

We introduced four types of students' learning activities and discussed possible usage and the way to collect. Our consideration mainly focused on how to collect learning activity data through Web-based system though. We are looking forward to discussing other types of activities and the way to collect learning activity logs from educational context not limited to programming environment.

REFERENCES

Blikstein, P. (2011). *Using Learning Analytics to Assess Students' Behavior in Open-ended Programming Tasks*. Paper presented at the First International Conference on Learning Analytics and Knowledge.

Fu, X., Shimada, A., Ogata, H., Taniguchi, Y., & Suehiro, D. (2017). *Real-time Learning Analytics for C Programming Language Courses*. Paper presented at the Seventh International Conference on Learning Analytics and Knowledge.

Taniguchi, Y., Okubo, F., Shimada, A., & Konomi, S. (2018). *How are Students Struggling in Programming? Understanding Learning Processes from Multiple Learning Logs*. Paper presented at the 11th International Conference on Educational Data Mining.