Pair programming learning support system for elementary school students that promote discussion

2310151 Hirono Makabi

Modern society is undergoing unpredictable changes due to technological innovation. Therefore, school education aims to develop children's ability to become creators of a better society and life in a changing environment. To achieve this goal, programming studies have become compulsory in Japanese elementary schools from 2020. The purpose of programming studies is to develop children's ability to use information technology, logical thinking skills, and the ability to collaborate and cooperate with others to solve problems. Therefore, in this study, we focused on "pair programming". Pair programming is a programming style in which two programmers work side by side at one computer and continuously collaborate on the same design, algorithm, code, and tests. Each pair is assigned to the role of driver and navigator. The driver inputs data into the computer and writes down the design. The navigator monitors the driver's work and looks for tactical and strategic flaws. Pair programming provides an environment where students can consult with others and work together to solve problems they do not understand, which may contribute to eliminating students' aversion to programming and cultivating the ability to collaborate and cooperate with others more than when they study programming alone. However, many previous studies have shown that many elementary school students do not ask questions even if they do not understand something, and that high-quality questions are not asked. In order to fully obtain the learning effects of pair programming, it is necessary to support elementary school students' discussions (questions). However, many of the existing support methods are unsuitable for pair programming learning or the immature developmental stage of elementary school students.

In this study, we therefore develop and introduce a learning support system that promotes discussions in pair programming learning for elementary school students and examine whether it can encourage more discussions and improve the quality of discussions among students, thereby contributing to improved learning outcomes. To achieve this, we designed a learning environment for pair programming and a method to support discussions. We then developed teaching materials and a web application to realize the design and conducted a demonstration experiment.

Dillion expresses the discussion (questioning) process in three steps: ① eliciting a sense of doubt, ② generating a question, and ③ expressing a question. In this study, we provide support for each of these three steps. Each support method is summarized below.

① By allowing learners to view the length of the discussion and the number of attempts

to answer, the navigator hints at when to ask questions.

② By providing a list of question stems (fill-in-the-blank questions) that correspond to students' lack of programming experience and the basics of discussion, learners can use these as an opportunity to ask questions.

③ The motivation of the learner to ask questions is increased by having the agent talk to them or by giving a certificate and praising the person who asks a question.

This study expressed these three support methods using a web application and cards. The completed system was then used to conduct a demonstration experiment (programming class) with 16 elementary school students, lasting a total of three hours. Three teachers (Teaching Assistants) who supported and monitored the students also participated in the experiment. Data collected during the experiment included questionnaires for students and teachers, comprehension tests, records of speech during pair programming, and learning logs for the web application. Based on the results of this analysis, the effectiveness of the support methods used in this experiment is summarized below.

① The learning log viewing function received many positive responses in the student questionnaire. In addition, some pairs expressed concern about the time it took to answer. However, a questionnaire by the teacher revealed that the roles of programming (navigator/driver) were unclear.

② The question stems for programming learning were not used much by the students, so it was not possible to confirm whether it improved the learning effect. The results of the teacher questionnaire and analysis of the content of their utterances revealed that while the students asked many simple questions during learning, such as "Why are not you getting the answer right?", these did not progress to asking deeper questions that questioned the intention of the program. There were also times when children tried to use a card when they got stuck, but the person they were asked could not answer and got stuck.

③ The results of the children's questionnaire suggested that the system of awarding a certificate to those who asked good questions may affect children's motivation to ask questions. Although many children responded positively to the agent in the children's questionnaire, there was a lack of experimental data to examine the effect of the agent on the content of their speech.

We also investigated whether using the question stems would change learning performance or speech, but because the question stems were not used much, no changes were observed. It was also suggested that pairing up children with different levels of confidence in talking with others would result in differences in the amount of speech during programming exercises.

There are several areas for improvement in this experiment. First, many children who participated in the experiment were good at discussions and asked questions even without the question stems. Therefore, the purpose of the question stems needs to be changed from "a trigger for questions" to "a way to ask advanced questions necessary for problem solving." For example, it would be good to use questions that ask about the details of the program. However, more than half of the children who participated in this experiment answered in the questionnaire that they were good at discussions. Since children who are not good at discussions may need “question stems that aim” or “a question stem that aims” to "provoke questions," it is necessary to consider appropriate question stems according to the differences in children's personalities and abilities. Second, by having children practice pair programming roles and questions before learning pair programming, they may be able to perform programming roles and questions more effectively. This is because teachers responded in the questionnaire that "the students were so engrossed in solving the problems that they forgot about the programming roles and question stems," and "some students seemed confused about how to use the cards." By having students practice thoroughly in advance, it may be possible to reduce the burden on them during learning and have them behave and ask questions according to their roles in pair programming more smoothly. Third, the results of this experiment did not clarify whether the question stems are effective in encouraging deeper learning content, or whether the questions are being asked by the agent's utterances. Therefore, additional investigation into these issues is necessary.

Finally, 80-90% of the children who participated in the experiment were in the fourth grade of elementary school. More data from fifth and sixth graders would be needed to improve the reliability of the data. In addition, because the author analyzed the utterances alone, the classification results should ideally be examined by multiple highly reliable people to determine whether they are correct.