Rodrigo, M. M. T. et al. (Eds.) (2021). Proceedings of the 29th International Conference on Computers in Education. Asia-Pacific Society for Computers in Education

A Framework of Exercise Recommendation for Novice Learners in Computer Programming

Lalita NA NONGKHAI[a*], Jingyun WANG[b] & Takahiko MENDORI[a]
[a]*Graduate School of Engineering, Kochi University of Technology, Japan*
[b]*Department of Computer Science, Durham University, UK*
*256010u@gs.kochi-tech.ac.jp

Abstract: To support novice learners majoring in Information Technology, this paper proposed an ontology-based framework to recommend exercise to learners in response to their understanding levels. This framework involves three essential mechanisms: (1) determination of the level of exercise, (2) evaluation of learner's understanding, and (3) recommendation of exercise. The prototype system will be implemented to enable the instructors to add or edit exercises based on 60 questions with solutions covering 20 topics of introductory programming which are initially determined by this research.

Keywords: Adaptive learning, practice recommendation, ontology, supervised learning

1. Introduction

In recent years, adaptive learning has been widely adopted in e-Learning (Paramythis et al., 2004) due to its mechanism which aims to enhance the learning efficiency by making use of the learner's characteristics. It has been proven that presenting the lesson in response to the learner's characteristics can help learners gain a better understanding (Ai et al.,2019, Seounghun,2021). Some adaptive learning (Paramythis et al., 2004) focused on providing adaptive interaction, such as personalized graphics or color, font style, or font size on content. Others focused on adaptive course delivery in response to the learner's characteristics.

For problem-solving related subjects, such as Mathematics, Sciences, and computer programming, they involve some content which is difficult to be taught with the same teaching strategy or material. For instance, many novices without a background in programming skills struggle with the programming course, the introductory course for undergraduate students majoring in Information Technology. Even after learning the programming concept after lectures, some of them still cannot solve relevant programming problems. It is no surprised that the learners who do not understand the lecture content struggle with the practical. However, some learners, who believe they understand the programming concept in the lecture, also cannot use the corresponding knowledge to solve the programming problem in a practical class. The fundamental way to improve programming skills is to practice more programming problems (Baboucar et al., 2019). The general approach to assign practice to learners is to design an exercise from easy-level to difficult-level. However, this approach is not appropriate for some learners because they have difficulty in locating the next practice when they cannot solve one specific. In the practical class, if the learners get stuck with one problem, the instructor can suggest they relocate to another easier one (or refer to prior knowledge); for this situation, they can solve the program with the help of the instructor. In cases of outside the classroom, it is difficult for the learners to solve this problem without the help of the instructor.

In this paper, we present an approach to improve programming skills for novice learners by recommending the exercise in response to the ability or understanding. Firstly, we collected programming questions with solutions addressing basic knowledge concepts in programming and divided those questions into three levels: easy, medium, and difficult. Furthermore, we design an ontology-based framework of programming exercise recommendations for novice learners. We plan to implement a system based on this framework and examine the effectiveness of its programming practice recommendation mechanism integrating the supervised learning model and domain ontology.

2. Literature Reviews

There are several researchers used Bloom's Taxonomy to define the exercise level. For instance, Seounghum et al. (2021) proposed a method to define the level of programming exercise with Bloom's Taxonomy. They used four from six cognitive learning skills in Bloom's Taxonomy and paired those four into two categories; "Remember & Understanding" and "Applying & Analyzing." Their exercise required the learners to understand a concept of programming such as "What is a data structure that follows the FIFO method?". However, our research target requires the learners to solve the problem by using programming skills. Zahid Ullah et al. (2020) also suggested assessing learner competencies based on Bloom's Taxonomy to decide which cognitive skills is appropriate to define the programming exercise level. They found the relationship between related cognitive learning skills that assess by PLS-SEM Model. The topic of programming in each cognitive included Selection, Iteration, and Modular. They provided the exercises that followed their cognitive skill definition and used the average score to assess that cognition. Other than only three topics, our research target covers all the basic programming topics and we develop a recommendation system to recommend exercise for the learners.

Baboucar et al. (2019) proposed to use ontology to describe programming exercise in Pascal Language into three levels; easy, medium, and difficult. The levels of exercise were defined on the general teaching sequence in basic programming, such as easy level which included display, variables, or simple selection, and medium level which included data structures or function. That means the learners who are provided with an easy level will not receive the exercise of the data structure or the function. In contrast, our research focuses on defining the levels of programming exercise with a logical increment in the concept based on ontology information instead of general teaching sequence in programming.

Taguchi et al. (2004) proposed the recommendation algorithm based on collaborative filtering. They design the exercise sequence step by step depending on the process of learning. They used collaborative filtering in two ways. On the one hand, they predict the score for new learners by the score of another learner with similar behavior and provide the new sequence of programming exercises based on their score and the score of another learner. On the other hand, they considered the student’s self-evaluation by questionnaire after practicing the exercises. They define the learner level into five groups: (1) solving a problem confidently, (2) understanding almost everything, (3) practicing by themself but not sure about the solution, (4) solving the problem by asking the instructor or friends, and lastly, (5) not completing exercise by themself. Different from this recommendation methodology, our research recommends the exercise depending on individual learning history.

Ai et al. (2019) proposed the exercise recommendation based on the deep reinforcement learning method using a multi-armed bandit algorithm, deep knowledge tracing, and SPARFA framework, a novel framework for machine learning based on learning analytics. This algorithm tracks learners with their learning history to recommend the exercise in response to their ability by using the deep reinforcement method. Furthermore, it uses the reward policy to drive the learners to practice a subsequence exercise when they get more than the threshold and to cheer them up when their score is not up to the threshold.

3. A framework of programming exercise recommendation system

A framework as shown in Figure 1, is designed to recommend programming exercises to novice learners majoring in Information Technology in response to their ability. This framework includes the process of defining the programming exercise levels, the process of evaluation of the learner's understanding levels, and the process of exercise recommendation.

3.1 Question Bank

We firstly designed a question bank that included around 20 topics of programming: display statement, variable, calculation, input data, if-statement, if-else, if-else-if, For Loop, While Loop, Nested Statement, Break, Continue, Define-Method, Standard-Method, Array, For-Each, List, Dictionaries, Try-Catch, and File. Two experts who have programming teaching experience for years designed 60 questions with solutions related to those topics and each topic includes easy, medium, and difficult levels. Each topic may cover several knowledge concepts defined by domain ontology describe a level of a programming exercise. Our research combines Bloom's Taxonomy with the domain ontology to define the level of exercise. Figure 2 shows an example of a question related to the topic of iteration and its solution provided in three programming languages (Python, Java, and C#).

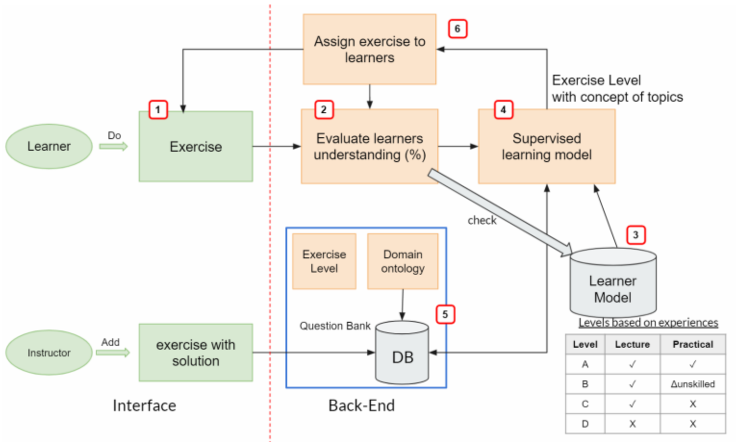

Figure 1. A Framework of Programming Exercise Recommendation System for Novice Learners.

11. Iteration 10 numbers (Learning Point: For Loop)
Write program to display the first 10 natural numbers.

```
Sample Output
-----------------------------------------
1 2 3 4 5 6 7 8 9 10
```

Python

```
a = ""
for x in range(1, 11):
    a += (str(x) + " ")
print(a)
```

Java

```
public class Test {
    public static void main(String[] args) {
        for (int i = 1; i <= 10; i++) {
            System.out.print(i + " ");
        }
    }
}
```

C#

```
using System;
namespace C_Sharp_Code {
    class Program {
        static void Main(string[] args) {
            for (int i = 1; i <= 10; i++)
                Console.Write(i + " ");
        }
    }
}
```

Easy	Display	Variable	Calculation	Input	IF	IF-ELSE	IF-ELSE-IF	For Loop	For-Each	Method	Array	List	Dictionaries
1	/												
2		/	/										
3		/	/										
4		/	/	/									
5		/	/	/									
6		/	/	/									
7		/		/									
8		/		/		/							
9						/							
10							/						
11								/					
12			/					/					
13			/					/					
14									/		/		
15					Nested			/		/			
16										/		/	
17			/		Nested			/		/			
18											/		
19			/					/	/				/
20					/								/

Figure 2. Example Question and Solution.

3.2 Interfaces

3.2.1 Instructor Interface

An instructor interface will be implemented to provide the function for the instructor to manage the question bank. The question bank will be stored in a database (item 5, figure 1). The instructor will be able to add an exercise with a solution into the database. Exercise level and knowledge domain defined by ontology will be the metadata of the added questions. The exercise recommendation process is to choose the exercise in response to a learner's level and assign that exercise to the learner.

3.2.2 Learner Interface

At first, the learners are required to do a pretest about programming and a questionnaire to evaluate their understanding. This information will be the initial data for the learner model (item 3, figure 1). For this learner model, we define a level of learners based on the work of Taguchi et al (2004). Their system used a questionnaire to divide learners into five groups, but the first and second groups defined in their research are very similar. Therefore, we combine them into one level as Level A. In other words, the four levels of the learners defined in our learner model are as follows.

- Level A: who understand the concept of programming (in lecture class) and solves program problem with confidence (in practical class).
- Level B: who understand the concept of programming and he/she take a lot of time to solve the programming problem (unskilled learners).
- Level C: who understand the concept of programming but cannot solve the programming problem such as they do not know how to solve the problem.
- Level D: who do not understand the concept of programming and cannot solve the problem.

The learner model will be used in the exercise recommendation process that includes the mechanism to recommend appropriate exercise to learners depending on the ability or understanding. After learners submit the solution of an exercise, the system will evaluate a learner's solution and then re-evaluate the learner's understanding for recommending the next exercise. This process may finish until the learner's understanding is more than a threshold and she/he can continue with the next topics in programming.

3.3 The process of exercise recommendation

The exercise recommendation mechanism in this framework makes use of a supervised learning model to find the next exercise for learners, as shown in Item 4, Figure 3. The concepts described in the domain ontology and the real-time learner model based on the learning history consist of the input of our supervised learning model. As the output of this supervised learning model, the recommended exercise will be determined based on the difficulty level and the concepts in the knowledge domain that learners are not understanding.

4. Conclusion and Future work

In this paper, a question bank including 60 questions about 20 topics related to knowledge concepts of programming with the solutions are designed. Since the criterion of the difficulty level of all the 20 topic is decided by referring to those 60 existing questions, we are working on extending the question bank to at least 4 times to current size, in total 240 questions. Moreover, a framework is proposed to recommend programming exercises to novice learners in Information Technology with three processes: determination of the levels of exercise, evaluation of learner's understanding, and recommendation of the exercise. In future work, a domain ontology will be designed to support the determination of the metadata of the programming exercises. Also, the supervised learning model will be implemented to suggest the suitable exercises based on the individual ability.

References

Paramythis A., & Susanne, L. R. (2004). Adaptive Learning Environment and e-Learning Standards. *Electronic Journal on e-Learning*, *2*(1), 181 – 194.

Seounghun, K., Woojin, K., & Hyeoncheol, K (2021). Learning Path Construction Using Reinforcement Learning and Bloom's Taxonomy. *The 17th Conference of the series of Intelligent Tutoring System Conferences*. Athens, Greece, June 2021.

Zahid, U., Adidah, L., & Mona, J. (2020). Bloom's taxonomy: A beneficial tool for learning and assessing students' competency levels in computer programming using empirical analysis. *Computer Applications in Engineering Eudcation*, *26*(6), 1628 – 1640.

Baboucar, D., Adrien, B., & Samuel, O. (2019). PasOnto: Ontology for Leaerning Pascal Programming Language. *2019 IEEE Global Engineering Education Conference (EDUCON)*. Dubai, United Arab Emirates, April 2019.

Ai. F., Chen. Y., Guo. Y., Zhao Y., Wang Z., & Fu, G. (2019). Concept-Aware Deep Knowledge Tracing and Exercise Recommendation in an Online Learning System. *The 12th International Conference on Education Data Mining*. Montreal, Canada, July 2019.

Taguchi, H. & Shimakawa, H. (2004). Exercise Sequence Adaption in Programming Education. *International Symposium on Communications and Information Technology 2004*. Sapporo, Japan, October 2004.