Interactive Chatbot for Supporting Students in Online Python Programming Class

Purit Phanudom[1,a)] Toshiki Hirao[1,b)] Raula Gaikovina Kula[1,c)] Hajimu Iida[1,d)]

Abstract: Learning programming language online gains popularity and sees realization in many variants. This includes massive open online course (MOOC), such as Codecamp[a] or Codecademy[b], or those in educational curriculum as part of programming subject with assigned teaching assistants, shortly known as TAs. However, having these TAs answer students' questions or help them solve programming errors may not be sufficient enough in terms of responsiveness, scalability, and coverage of answers given back to students. This paper introduces the idea of applying chatbot technology to support and interact with students during online lessons, and presents the preliminary results from this study in progress.

[a] https://codecamp.jp/
[b] https://www.codecademy.com/

Keywords: Online Education Support, Error Message Enhancement, Chatbot

1. Introduction

In online classes, teaching assistants (TAs) support students to solve program errors real time. Its limitation is when TAs are not able to handle students' requests fast enough. More importantly, how much of usefulness the TAs' responses also depend on level of expertise of programming of the TAs. This paper introduces the concept of applying interactive chatbot whose aim is to provide solutions for incoming questions of the students. The chatbot is planned to be embedded into C2Room [1] (i.e., an online collaborative editor for Python programming provided by dTosh Inc.[*1]), and the proposed solution will be compared against baseline TAs as well as other alternative learning support solutions.

The rest of this paper is organized as followed. Section 2 discusses related works, followed by Section 3 outlining the plans for implementation and conducting validation experiment. Next is the preliminary results in Section 4, then the conclusion and future works in Section 5.

2. Background and Related Work

Studies related to our research can be broadly grouped into three aspects - studies on error message enhancement, quantification of error enhancement performance, and chatbot application in education environment.

[1] Nara Institute of Science and Technology, Ikoma, Nara 630–0101, Japan
[a)] phan-udom.purit.pm6@is.naist.jp
[b)] hirao.toshiki.ho7@is.naist.jp
[c)] raula-k@is.naist.jp
[d)] iida@itc.naist.jp
[*1] https://www.dtosh.com/

2.1 Enhancing Compiler Error Messages

In attempts to help students comprehend error messages, there have been many studies that concentrate on delivering 'enhancement' of what are the original compiler error messages. These enhancements serve as supplementary explanatory information of the original error message in more understandable and intuitive manners.

Expresso [2] scans for 20 common Java compiler errors and provides fixing guides to the students; however its true efficiency was left unanswered. Next is Decaf [3], a special IDE for Java that present compiler message along side with original error messages. Another tool named PyCee [4] introduces the use of accept answer from relavant StackOverflow threads to support explanation of occurred errors in Python compiler, installed as a plugin for Sublime editor.

2.2 Measuring Effectiveness of Error Message Enhancement

To quantify how effective the solutions for supporting students' learning are, some metrics have been proposed to indicate the degree of student's learning outcome in the context of error messages.

Number and frequency of repeated error have been regarded as a significant indicator in how well students learn from error messages [5]. Claiming to be less context-dependent than the pre-existing Error Quotient metric (EQ) [6], Becker suggested a new metric called Repeated Error Density (RED) [7] that analyse the sequence of same consecutive errors generated.

2.3 Chatbot for Education Support

A systematic study [8] has revealed a number of studies with

```
{
    "tags":"ZeroDivisionError",
    "patterns":[
        "ZeroDivisionError: division by zero",
        "ZeroDivisionError: integer division or modulo by zero"
    ],
    "responses":[
        "It is not possible to perform division using zero as
        denominator, or module with zero.",
        "Make sure that the denominator, or the modulo number,
        is not equal to zero."
    ]
},
```

Fig. 1 Structure of chatbot intent in JSON notation format

similar objective such as ours. One of them is the University of Georgia's 'Jill Watson' chatbot [9] whose role was to handle forum posts by students enrolled in a computer science course. Differing slightly from chatbot are the pedagogical agents and intelligent systems, such as the work of Goettl et. al [10].

3. Methodology

The plan of our study begins with data collection and processing, whose final format would be used to implement the actual chatbot for C2Room, and the validation experiment to investigate the impact of chatbot on students' learning improvement.

3.1 Data Collection and Processing

The compilation output of students' source code has been recorded in C2Room system. The log data also includes error messages if the compilation fails. Along with the data set of discussion message between students and TAs, altogether these will be used to build the chatbot to interact and response with similar patterns as regular TAs.

3.2 Coding Support in C2Room

C2Room allows students to submit their source code for revisions, as well as directly chat with anonymous TAs about the exercises. This is particularly common when students are unable to figure out the cause of errors. It is expected that the chatbot will be a replacement to TAs, if not available as a setting option.

Table 1 Types of compiler error messages and their number of unique patterns

Exception category	Number of Unique Message Patterns
AttributeError	5
EOFError	1
FileNotFoundError	1
ImportError	2
IndentationError	3
IndexError	6
KeyError	1
ModuleNotFoundError	1
NameError	1
RecursionError	1
RuntimeError	1
SyntaxError	24
TabError	1
TypeError	73
UnboundLocalError	1
ValueError	50
ZeroDivisionError	2
Other*	7

*These errors do not fall into main types, and belong to external modules or framework (i.e. numpy.AxisError)

3.3 Validation Experiment

To measure how effective our proposed solution is, controlled experiment against other error message enhancement tools will be conducted along with baseline group. Every student group will be assigned with the same set of coding exercises. However, each will be provided with different mean of assistance; regular TAs, chatbot, and compiler messager enhancer (i.e. PyCee).

4. Preliminary Results

From the initial data set, types of errors (exceptions), and their unique variants have been identified. Table 1 lists the types of exceptions and number of variants from the data. Based on the results obtained, exceptions which have the significant diversity in the error messages are `TypeError`, `ValueError`, and `SyntaxError`. Figure 1 shows how an intent is structured and defined for chatbot training in JSON notation. Currently this is still a prototype version. Further refinement and completion with other compiling exceptions is expected.

5. Conclusions and Future Works

This paper presents a study in progress on utilizing chatbot technology as an interactive assistance for students taking online Python class. The dataset reveals characteristics and types of errors encountered, and how the TAs response to students to support solving errors. Future works are to realise the implementation using the processed data, and carry out evaluation study that can measure the efficiency of the chatbot solution in comparison to existing conventions.

Acknowledgments This work was supported by JSPS KAKENHI Grant Number JP21H03416.

References

[1] dtosh online class system c2room. https://www.c2room.jp, 2021. accessed 8 August 2021; Online.

[2] Maria Hristova, Ananya Misra, Megan Rutter, and Rebecca Mercuri. Identifying and correcting java programming errors for introductory computer science students. *ACM SIGCSE Bulletin*, 35(1):153–156, 2003.

[3] Brett A Becker, Graham Glanville, Ricardo Iwashima, Claire McDonnell, Kyle Goslin, and Catherine Mooney. Effective compiler error message enhancement for novice programming students. *Computer Science Education*, 26(2-3):148–175, 2016.

[4] Emillie Thiselton and Christoph Treude. Enhancing python compiler error messages via stack. In *2019 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*, pages 1–12. IEEE, 2019.

[5] Matthew C Jadud. *An exploration of novice compilation behaviour in BlueJ*. PhD thesis, University of Kent, 2006.

[6] Matthew C Jadud. A first look at novice compilation behaviour using bluej. *Computer Science Education*, 15(1):25–40, 2005.

[7] Brett A. Becker. A new metric to quantify repeated compiler errors for novice programmers. In *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education*, ITiCSE '16, page 296–301, New York, NY, USA, 2016. Association for Computing Machinery.

[8] Rainer Winkler and Matthias Söllner. Unleashing the potential of chatbots in education: A state-of-the-art analysis. 2018.

[9] Ashok Goel, Brian Creeden, Mithun Kumble, Shanu Salunke, Abhinaya Shetty, and Bryan Wiltgen. Using watson for enhancing human-computer co-creativity. In *2015 AAAI fall symposium series*, 2015.

[10] Barry P Goettl, Henry M Halff, Carol L Redfield, and Valerie J Shute. *Intelligent Tutoring Systems: 4th International Conference, ITS'98, San Antonio, Texas, USA, August 16–19, 1998, Proceedings*. Springer, 2003.