Yang, J. C. et al. (Eds.) (2018). Proceedings of the 26th International Conference on Computers in Education. Philippines: Asia-Pacific Society for Computers in Education

Comment Generation by LSTM for Procedure Learning System

Akiyoshi TAKAHASHI[a], Hiromitsu SHIINA[b*] & Nobuyuki KOBAYAHSHI[c]
[a]*Graduate School of Informatics, Okayama University of Science, Japan*
[b]*Faculty of Informatics, Okayama University of Science, Japan*
[c]*Faculty of Regional Management, Sanyo Gakuen University, Japan*
*shiinahiromitsu@gmail.com

Abstract: For beginners of the program, we are developing a system that learns the sequence of program procedures on a tablet computer. We would like to create many exercises for the procedure learning system. Therefore, a system for automatically generating procedures is needed. In this study, in order to generate a procedure from source code, learn source code and corresponding comment using Encoder-Decoder model implemented by LSTM of neural networks and generate comment corresponding to new programs. Also, when targeting learners who are a little familiar with the program, we also show that we can deal with preprocessing comments corresponding to the program.

Keywords: Algorithm learning, Procedure learning, Procedure generation, Neural networks, LSTM

1. Introduction

In the progress of the information society, it is required to train information utilization abilities. In school education, it is necessary to enrich at all educational stages of ICT education such as programming learning. Preparations have already started in elementary, middle, and senior high schools, including making programming education compulsory in elementary schools by 2020 in Japan (MEXT, 2018). Therefore, it is necessary to consider the programming education for university students and the connection of ICT education at elementary, junior high and high school, so it is necessary to prepare content for programming learning and study its support system. In particular, in ICT education, it is important to operate programs directly on computers, but "Computational thinking" (Wing, 2006) is more important than program grammar learning.

In this study, the purpose is to aid understanding of the procedure to solve the problem. Thus far, we have constructed a procedure learning system that rearranges algorithm procedures on a tablet PC. In the procedure generation of the learning system, since it is troublesome for the lecturer of the problem and wanting to create many exercises, a system for automatically generating the procedure by the computer is necessary. Since procedures are similar to comments, we consider that comments and procedures can be generated automatically. With the development of a method using neural networks, natural language translation and sentence generation is developing. In particular, an Encoder-Decoder translation using LSTM has been proposed, and translation accuracy has been improved. In this study, pairs of one line of program list and its comments are learned in the Encoder-Decoder model used for translation from program to comments. Also, the LSTM generates a comment for a new program list. In Computer Thinking learning, we think that it is necessary to be able to summarize and disassemble the explanation of the algorithm procedure. It is not necessary to be conscious of variables in the first stage of learning. However, it is necessary to be conscious of the programming language in the second stage. In particular, it is necessary to know about the change of the variable. Also, since the information on the variables is insufficient only with the original program list, the flow relating to the variables of the program tends to be difficult to understand. Therefore, we added conversion of variable information and complement variable name.

2. Procedure learning system

Programming involves several procedures such as declaring variables, inputting values, calculation, and the output of results. This programming procedure test has a procedure for converting the programming into Japanese and dividing that procedure into several parts. There is the text that rearranges the divided procedures in order from the top so that they match the flow of the programming. This test is utilized for understanding the learning situation in lectures(Figure 1).

Figure 1. Procedure learning system.

3. Comment generation with LSTM

3.1 Outline of Comment generation system by Encoder-Decoder translation model

There is research into the use of an Encoder-Decoder translation model using a deep learning LSTM to translate. In this study, we consider the translation model as a model to translate from the program list to comments and attempt to generate comments. The following process shows comments generate process with information of variables, and these processes show in Figure 2.

(1) Separate source programs with comments into programs and comments.
(2) Retrieve of a variable name, and generate comments with a variable name.
(3) Learn to LSTM a pair of programs and comments in parallel.
(4) LSTM generates new comments from new programs without comments.
(5) Processing for changing the variable name of the comment,

3.2 Learning pairs of program and comment to LSTM in Encoder-Decoder translation model

We use Encoder-Decoder model composed of LSTM. Preprocessing to LSTM, the structure of LSTM and learning are shown following description, and right side of Figure 2 shows the structure of LSTM.

(1) Preprocessing to LSTM: In order to learn the Encoder-decoder translation model, input sentences need to separate for each word. In other words, the program source code is divided into a token sequence, and Japanese comments are divided into word sequence of morphemes. The LSTM model uses a token sequence of program source code and word sequence of comments for input.

(2) Structure of LSTM: In the LSTM block, the error from the previously stored data in the parallel data of the program and comment is a loss. The losses are accumulated in each LSTM block. Finally, learning of parameters is performed from the total loss by error backward propagations.

(3) Learning pairs of program and comment in LSTM: In the LSTM data learning, the LSTM learns the connections of words for each line of a program. Programs divided into tokens are input token by token into the LSTM, and context information is accumulated. After that, each word of comments is input to the LSTM, and the LSTM learns parameters by generating words.

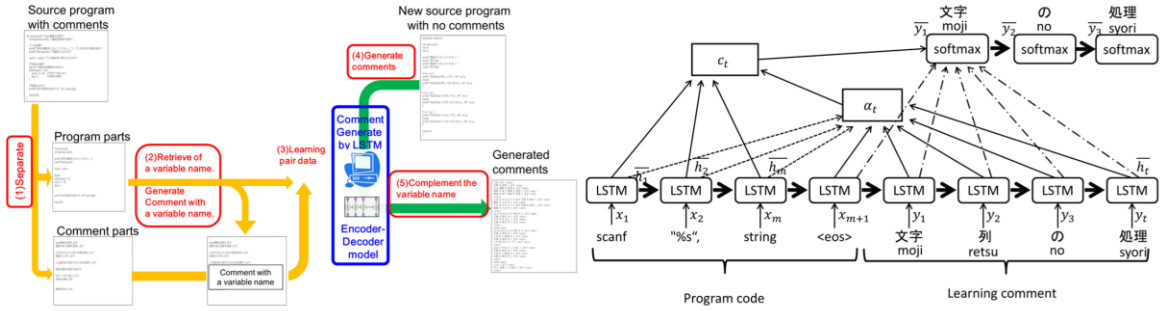

Figure 2. Comments generation system overview and learning a pair of program list and comment with LSTM.

4. An example of Comment generation process

The LSTM comment generation process for a new program list without comments is show in Figure 3. In the LSTM, the comment generated by the learning number of learning data of programs and comments pair is different. The generated comments by LSTM learned 100 times and 500 times are shown in the center of Figure 3. We use generated comments in 500 times iteration.

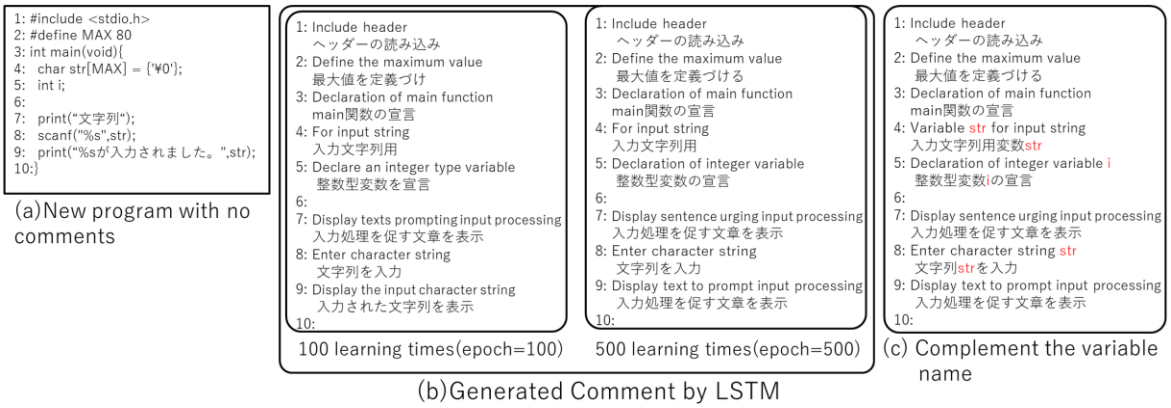

Figure 3. An example of comment generation process

5. Conclusion and Future works

Comment generation by simple learning of pairs of program list and comments has been successful in the part of fitting learned comments. In the questionnaire, although evaluation of Japanese generation is good, problems are pointed out because there are cases where procedures are generated for different meanings. New terms and the like not included in learning cannot be generated. As for a future task using structure information is considered to be an important improvement. Also, we would like to extend the information of the lecture to the LSTM comment generation.

References

MEXT. (2018). Elementary school programming education guide (1st edition). Retrieved August 16, 2017 from http://www.mext.go.jp/a_menu/shotou/zyouhou/detail/1403162.htm (in Japanese)

Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, *49*(3), 33-35.

Sutskever, I. Vinyals, O. Le, Q. V. (2014). Sequence to Sequence Learning with Neural Networks. *Advances in Neural Information Processing System*, *27*(NIPS 20142), 3104-3112.