Shanghai Jiao Tong University

HW5: Learning From Programs

Student name: *520030910342 Jiyu Liu*

Course: *Data Mining* – Professor: *Liyao Xiang*

Date: *6.10 2023*

1 Abstract

In homework5, I have completed two programs in the field of source code learning. The first one is to use Abstract Syntax Tree (AST) parser to extract code features, while the second one is to implement a simple code search model.

2 Program One

In program one, I implement three functions to extract method names, object creation expressions and object invocation expressions respectively.

In the function of extracting method name, I iterate over every child of class body, find those with the type of 'method_declaration', then decode the second child (identifier) of the node and add it to the method names list.

In the function of extracting object creation expressions, I iterate over all children of class body and use a recursive method to find all nodes with the type of 'object_creation_expression'.

In the function of extracting object invocation expressions, at first, I use similar recursive searching methods to find all nodes with the type of 'method_invocation' and 'local_variable_declaration' separately. Second, I use the local variable declaration list to create a dictionary that maps every identifier to its corresponding type identifier. Finally, I replace objects of method invocations which are identifiers with their types.

3 Program Two

In program two, I implement the code search model based on the description and code release of [1]。The input code features can be divided into three types: name, token and description.

ACC@10	MRR	NDCG@10
0.47917	0.18214	0.25189

表 1: Best Result

For name input and description input, I use embedding layer, LSTM model and max-pooling layer to get the encoding output. For token inputs, I use embedding layer and max-pooling layer to get the encoding output. Finally, I concatenate the name encoding and token encoding and pass it into a linear layer to get the code representation, and pass description encoding into another linear layer to get the description representation. In epoch 13, my model achieved the best performance, which can be seen in table 1.

References

[1] Xiaodong Gu, Hongyu Zhang, and Sunghun Kim. "Deep Code Search". In: *Proceedings of the 2018 40th International Conference on Software Engineering (ICSE 2018)*. ACM. 2018.