学 位 論 文 概 要

題　目　　A study of combining code representation techniques with deep learning for software defect prediction（ソフトウェア欠陥予測のためのコード表現技術とディープラーニングの組み合わせに関する研究）

氏　名　　FANG DINGBANG

To improve the performance of the model, this dissertation accordingly proposes three different solutions to improve the performance of defect prediction, as follows:

a) As the complexity of software makes programs difficult to understand, appropriate representations that capture features from different levels of abstraction in the code can effectively represent the information in the code. We introduce *Gated Homogeneous Fusion Network* (GHFNet) for defect prediction, combining high-level semantic feature extraction and weighted static feature extraction. Through the mechanism of homogeneous gating fusion, weights are adaptively assigned to the two types of features based on the correlation of these features to form fused features for defect prediction in the code. Experiment results show that the proposed algorithm learns multiple levels of features efficiently and outperforms the reference algorithm for defect prediction.

b) Some researchers have leveraged deep learning (DL) to learn semantic features from the abstract syntax tree to identify potential defects. However, they directly serialize the nodes in the abstract syntax tree to form a sequence as DL input, ignoring the structural information of the tree. To solve the above problem, we propose a property-enhanced lightweight graph (PLG) based on an abstract syntax tree (AST) reflecting structural information in the source code. The PLG, which retains nodes related to program semantics, is more lightweight than the AST as well as enhances the strength of the connection between leaf nodes and the parent nodes of semantically stronger attributes. PLG is expected to significantly reduce the complexity of the original tree and enhance the attribute relationships between nodes. The experiment results show that the proposed model has significant improvements for reference methods in several Java repositories.

c) A deep learning system (DLS) developed based on one software project for defect prediction may well be applied to the related code on the same project but is usually difficult to be applied to new or unknown software projects. To address this problem, we propose a Transferable Graph Convolutional Neural Network (TGCNN) that can learn defects from the lightweight semantic graphs of code and transfer the learned knowledge from the source project to the target project. We discuss how the semantic graph is constructed from code; how the TGCNN can learn from the graph; and how the learned knowledge can be transferred to a new or unknown project. We also conduct a controlled experiment to evaluate our method. The result shows that despite some limitations, our method performs considerably better than existing methods.