Engineering Village™

1. **Modeling Hierarchical Syntax Structure with Triplet Position for Source Code Summarization**
Guo, Juncai; Liu, Jin; Wan, Yao; Li, Li; Zhou, Pingyi **Source:** *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, v 1, p 486-500, 2022, *ACL 2022 - 60th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*; **ISBN-13:** 9781955917216; **Conference:** 60th Annual Meeting of the Association for Computational Linguistics, ACL 2022, May 22, 2022 - May 27, 2022; **Sponsor:** Amazon Science; Bloomberg Engineering; et al.; Google Research; Liveperson; Meta; **Publisher:** Association for Computational Linguistics (ACL)

Author affiliation:
School of Computer Science, Wuhan University, China
School of Computer Sci. and Tech., Huazhong University of Science and Technology, China
Faculty of Information Technology, Monash University, Australia
Noah's Ark Lab, Huawei, China

Abstract:
Automatic code summarization, which aims to describe the source code in natural language, has become an essential task in software maintenance. Our fellow researchers have attempted to achieve such a purpose through various machine learning-based approaches. One key challenge keeping these approaches from being practical lies in the lacking of retaining the semantic structure of source code, which has unfortunately been overlooked by the state-of-the-art methods. Existing approaches resort to representing the syntax structure of code by modeling the Abstract Syntax Trees (ASTs). However, the hierarchical structures of ASTs have not been well explored. In this paper, we propose CODESCRIBE to model the hierarchical syntax structure of code by introducing a novel triplet position for code summarization. Specifically, CODESCRIBE leverages the graph neural network and Transformer to preserve the structural and sequential information of code, respectively. In addition, we propose a pointer-generator network that pays attention to both the structure and sequential tokens of code for a better summary generation. Experiments on two real-world datasets in Java and Python demonstrate the effectiveness of our proposed approach when compared with several state-of-the-art baselines.
 (46 refs.)

Main Heading: Semantics **Controlled terms:** Abstracting - Computational linguistics - Natural language processing systems - Python - Syntactics - Trees (mathematics)

Uncontrolled terms: Abstract Syntax Trees - Automatic codes - Hierarchical structures - Learning-based approach - Machine-learning - Natural languages - Semantic structures - Source codes - State-of-the-art methods - Syntax structure

Classification Code: 721.1 Computer Theory, Includes Formal Logic, Automata Theory, Switching Theory, Programming Theory - 723.1.1 Computer Programming Languages - 723.2 Data Processing and Image Processing - 903.1 Information Sources and Analysis - 921.4 Combinatorial Mathematics, Includes Graph Theory, Set Theory

Funding details: Number: 61972290, Acronym: NSFC, Sponsor: National Natural Science Foundation of China; Number: 62102157, Acronym: NSFC, Sponsor: National Natural Science Foundation of China;

Funding text: This work is supported by the National Natural Science Foundation of China under Grants 61972290. Yao Wan is partially supported by the National Natural Science Foundation of China under Grant No. 62102157. We would like to thank all the anonymous reviewers for their constructive comments on improving this paper.

Database: Compendex