Paper Review for CSE438

Samia Semu

ID: 23341136

Section: 01

Paper Title: Detecting Automatic Software Plagiarism via Token Sequence Normalization

Paper Link: https://dl.acm.org/doi/10.1145/3597503.3639192

1 Summary

1.1 Motivation

The paper introduces a novel defense mechanism for combating automated software plagiarism by normalizing token sequences effectively.

1.2 Contribution

The paper introduces a defense mechanism using semantic token enrichment, language-independent graphs, and topological sorting to counter obfuscation attacks effectively.

1.3 Methodology

The methodology enriches token sequences with semantic data to create language-independent graphs for normalizing tokens, removing dead statements, and establishing a fixed order. Topological sorting is utilized to reverse obfuscation effects, enhancing plagiarism detection effectiveness and resilience against automated attacks with minimal false positives.

1.4 Conclusion

The defense mechanism effectively combats automated obfuscation attacks, increasing similarity scores of plagiarized solutions with low false positives, providing a practical and reliable software plagiarism detection solution.

2 Limitations

2.1 First Limitation

While the defense mechanism effectively handles plagiarism generators that insert or reorder statements, it may face challenges with future complex attacks based on refactoring or reimplementation that change larger parts of the token sequence, potentially reducing its effectiveness in detecting such advanced obfuscation techniques.

2.2 Second Limitation

The evaluation was primarily conducted on Java datasets due to limited availability, indicating a potential limitation in generalizing the mechanism to other programming languages without extensive testing and adaptation, which could impact its applicability across a broader range of software development environments.

3 Synthesis

The paper's token sequence normalization approach has wide-reaching implications for enhancing plagiarism detection tools across programming languages. Future research could focus on further developing this method to counter evolving obfuscation techniques, incorporating deeper semantic analysis, and integrating uncertainty propagation for more robust defense mechanisms against emerging threats in software plagiarism.