Code signatures for plagiarism detection

Dennis Goßler Dennis Wäckerle

October 31, 2022

Abstract

Contents

1 Introduction

2 How Does Plagiarism Detection Work

Fand den Abschnitt bei moss ganz nett könnte man ja mit reinpacken Dennis G.

Moss and other plagiarism detection tools are not perfect, so a human should go other the results, and it should be checked if the clams are valid. "In particular, it is a misuse of Moss to rely solely on the similarity scores. These scores are useful for judging the relative amount of matching between different pairs of programs and for more easily seeing which pairs of programs stick out with unusual amounts of matching. But the scores are certainly not a proof of plagiarism. Someone must still look at the code." (Aiken, 2021)

3 Use Case and Software Experiment

3.1 Use Case

3.2 Software Experiment

4 Criteria for Evaluation

5 Evaluation of the different Tools

5.1 MOSS

Moss clams to be one of the best cheating detection algorithms. "The algorithm behind moss is a significant improvement over other cheating detection algorithms (at least, over those known to us)." (Aiken, 2021)

Moss supports the following programming languages C, C++, Java, C#, Python, Visual Basic, Javascript, FORTRAN, ML, Haskell, Lisp, Scheme, Pascal, Modula2, Ada, Perl, TCL, Matlab, VHDL, Verilog, Spice, MIPS assembly, a8086 assembly, a8086 assembly and HCL2

5.2 JPlag

5.2.1 JPlag's comparison algorithm

- Functions in two phases 1. all programs are parsed and converted into tokens 2. tokens strings are compared in pairs. Tries to cover one token stream with substrings of the other token. Percentage of covered token streams is the similarity. (Perchelt et al., 2000, p. 10)

- Tokenizing -> only language dependent process(Perchelt et al., 2000, p. 10) - Tokens represent syntactic elements e.g. statements or control structures(Sağlam, 2022, How are submissions represented? — Notion of Token)

- Transformation - each file is parsed -> result set of Abstract Syntax Trees for each submission - each AST traversed depth first, nodes are grammatical units of language - when entering and exiting a node a token can be created and added to the token list - block type node e.g. classes

or if expressions have corresponding begin and end tokens. Token list should have balanced pairs of matching begin and end tokens

- Comparing token strings

5.2.2 The results

5.2.3 Integration into an automated evaluation pipeline

5.3 Plaggie

5.4 AC2

The output that AC2 generates is visualization base. So AC it will not provide a value like "percentage of copy" instead, "it will create graphical representations of the degree of similarity between student submissions within a group" (Freire, 2022)

6 Conclusion

References

Aiken, A. (2021). *A system for detecting software similarity*. https://theory.stanford.edu/~aiken/moss/

Freire, M. (2022). *Ac*. https://github.com/manuel-freire/ac2

Perchelt, L., Malphol, G., & Phlippsen, M. (2000, March 28). *Jplag: Finding plagiarisms among a set of programs*.

Sağlam, T. (2022, September 16). *4. adding new languages*. Retrieved October 30, 2022, from https://github.com/jplag/JPlag/wiki/4.-Adding-New-Languages