

# Code signatures for plagiarism detection

Dennis Goßler

Dennis Wäckerle

October 31, 2022

## **Abstract**

## Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>How Does Plagiarism Detection Work</b>	<b>3</b>
<b>3</b>	<b>Use Case and Software Experiment</b>	<b>3</b>
<b>4</b>	<b>Criteria for Evaluation</b>	<b>3</b>
<b>5</b>	<b>Evaluation of the different Tools</b>	<b>3</b>
5.1	MOSS . . . . .	3
5.2	JPlag . . . . .	3
5.3	Plaggie . . . . .	3
5.4	AC2 . . . . .	3
<b>6</b>	<b>Conclusion</b>	<b>3</b>
	<b>References</b>	<b>4</b>

## 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 over the results, and it should be checked if the claims 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

## 4 Criteria for Evaluation

## 5 Evaluation of the different Tools

### 5.1 MOSS

Moss claims 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

### 5.3 Plaggie

### 5.4 AC2

The AC2 algorithm claims that it can be run locally. Other plagiarism detection tools require sending data to remote servers like MOSS. AC2 is using "Normalized Compression Distance" as its main measure-of-similarity, but with the possibility of integrating other additional measures.

The output that AC2 generates is visualization based. 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>