Abstract

In the domain of software engineering, detecting the use of open source code in obfuscated Android applications holds significant importance across various disciplines. The ability to identify syntactically or semantically similar code fragments, commonly referred to as clones, is essential in tasks such as code versioning, plagiarism detection and ethical considerations surrounding open source code usage.

This dissertation presents the concepts, artefacts and processes involved in the development of an APK Code Matching (ACM) application, designed to identify and match similar methods between open source and obfuscated variants of Java Android applications. The ACM employs a three-way approach, leveraging static code analysis, program call hierarchy, and the k-Nearest Neighbours (k-NN) algorithm for match detection of class methods.

The extensive evaluation carried out as part of the development process underscores the effectiveness and efficiency of the ACM application in terms of its ability to accurately map methods between APKs, even in scenarios involving complex and dense obfuscation.