Warsaw University of Technology

Faculty of Mathematics and Information Science

Abstract

Ph.D. Thesis

Title: A source code similarity assessment system for functional programming languages based on machine learning and data aggregation methods

Author: Maciej Bartoszuk

Supervisor: dr hab. inż. Marek Gągolewski, prof. PW

This thesis deals with the problem of detecting similar source codes (also known as clones) in functional languages, especially in R. Code clones' detection is useful in programming tutoring, where teachers and trainers wish to identify suspiciously similar works, as well as in assuring source code quality, so that repeated code is avoided.

This work introduces an operational definition of code similarity, which is used as a formal formulation of the problem: given a list of possible source code modifications, we say that two code fragments A and B are similar if A can be derived from B by means of these transformations. Based on the observation that currently employed assessment methods of clone detection algorithms are insufficient, a new approach to evaluate their performance and accuracy is proposed. Namely, the benchmark sets were created by randomly choosing functions, extracted from popular packages and transforming them by means of the aforementioned modifications. This allows to constrain the size of similar functions' clusters, as well as the fraction of clones and number of performed transformations. 10 random sets of functions generated according to 108 different parameter scenarios were considered.

One may note that the algorithms proposed so far use a fixed source code representation, e.g., tokens. There were no detailed studies on how accurate an algorithm would be if it was run against a different sequence, e.g., of raw characters. In this work, the algorithms, understood as methods to compare sequences or graphs, are separated from particular representations of source codes. By means of such an approach, the effectiveness of each algorithm could have been examined with respect to each possible code representation. In particular, a new way of code representation was proposed, namely the one that focuses on R function calls. In this study, four representations (three sequential and one graph-based) and seven comparison algorithms (four for sequential and three for graph

data) were examined.

The current state-of-the-art code clone detection approaches based on Program Dependence Graphs (PDG) rely on some exponential-time subroutines. In this work, the performance of a polynomial-time algorithm based on the Weisfeier–Lehman method for finding similar graphs is examined. What is more, its novel modification – the SimilaR algorithm has been empirically shown to outperform all the other inspected methods.

Since all the algorithms for comparing source codes focus on different code similarity aspects, the current work also ephasizes the need for a proper aggregation of the results generated by different approaches, especially in the context of solving the problem of binary classification and regression. However, one may note that many supervised learning algorithms like random forests or support vector machines, have the following disadvantage: one cannot easily quantify the influence of each particular method on the final result. Moreover, such models do not obey desirable properties such as the monotonicity or idempotence. What is more, they do not allow to compare the results of individual methods directly. In other words, e.g., a value of 0.76 returned by one of the algorithms does not necessarily represent the same degree of similarity as a value of 0.76 obtained using another method. Thus, a new similarity aggregation method (based on B-splines curves and surfaces) was proposed. The new models not only posses an intuitive interpretation, but also were experimentally shown to yield better results than, among others, the random forests algorithm.

It is also worth noting that the approaches presented in the literature rely on a symmetrical form of similarity: they return a single value describing how similar both functions are to each other. In this work, non symmetric measures were proposed, quantifying the degree to which one function is contained within a second one ($A \subset B$) and vice versa ($A \supset B$). Due to the appropriate aggregation of results (by means of t-norms (like the minimum, product and Łukasiewicz), the arithmetic mean or t-conorms (dual to the above-mentioned t-norms), several methods have achieved better performance than in the case of a symmetrical approach.

The thesis is accompanied by a publicly available open source R package published in the CRAN repository, as well as web service, both implementing the proposed algorithm.