

Using Structural and Semantic Information to Support Software Refactoring

- Abstract -

In the software life cycle the internal structure of the system undergoes continuous modifications. These changes push away the source code from its original design, often reducing its quality. In such cases refactoring techniques can be applied to improve the design quality of the system.

Approaches existing in literature mainly exploit structural relationships present in the source code, e.g., method calls, to support the software engineer in identifying refactoring solutions. However, also semantic information is embedded in the source code by the developers, e.g., the terms used in the comments.

This research investigates about the usefulness of combining structural and semantic information to support software refactoring. In particular, a framework of approaches supporting different refactoring operations, i.e., Extract Class, Move Method, Extract Package, and Move Class, is presented.

All the approaches have been empirically evaluated. Particular attention has been devoted to evaluations conducted with software developers, to understand if the refactoring operations suggested by the proposed approaches are meaningful from their point of view.