

# 要 旨

## ソースコード中の識別子の分析による リファクタリング支援法の研究

山本 風歌

プログラムの可読性や保守性を高めるために、プログラムの振る舞いを変えずに設計を見直す技術として、リファクタリングがある。様々なリファクタリング手法が提案されているが、本研究ではオブジェクト指向プログラミング言語における、インタフェースの抽出手法について着目する。既存の手法として、異なるクラスに存在する同名のメソッドを抜き出し、それらを新たなクラスとして定義する方法がある。しかし、この手法では類似するクラスの存在が抽出の前提条件であるため、現時点で類似する部分が存在しないソースコードからインタフェースの抽出を行うことが難しい。

そこで、ソースコード中に潜在するインタフェースを、既存ソースコードのクラス名とメソッド名の対応関係をデータマイニングの一手法である相関ルール抽出手法によって学習したデータと比較することで予測し、その情報を元にリファクタリングが行えるか検討する。具体的には、ある名詞を含む名前定義されたクラスのメソッドに頻出する動詞を、名詞が持つ抽象的な振る舞いとみなし、その対応関係を知識データとして収集する。リファクタリング対象のソースコード中に、知識データに収録された対応関係が存在する場合、インタフェースの抽出をプログラム作成者に提案する。

本研究では、過去に生産されたソースコード中のクラス名とメソッド名から、頻出する名詞と動詞の対応関係を抽出する方法について提案した。また、実験によって実際に対応関係が得られることを確認した。

キーワード リファクタリング, データマイニング, インタフェース抽出

# Abstract

## Support of refactoring via analysis of identifiers in source codes

Fuka YAMAMOTO

Refactoring is a technology to review the software design for improving readability or maintainability of the program. Various refactoring methods have been proposed, and this paper focuses on the extraction of interface. An existing approach extracts the same name method in different classes and defines a new class with that method. However, this approach is not applicable when two classes do not share the same name method.

This paper concerns a method for predicting a potential interface in source codes using the correspondence relation between class names and method names. Specifically, we collect the correspondence relation between nouns in class names and verbs in method names. If the correspond relationship is present in source code under refactoring, then the method suggests the extraction of interface.

This paper proposes a method for extracting frequently appearing correspondence relation between class names and method names in existing source codes. By performing experiments, it is confirmed that the proposed method can obtain such corresponding relationship.

*key words*    refactoring, data mining, interface extraction