

令和元年6月17日現在

機関番号：13901

研究種目：若手研究(B)

研究期間：2016～2018

課題番号：16K16034

研究課題名(和文) コードクローンの変化に対応するリファクタリング支援環境の構築

研究課題名(英文) Refactoring environment for the evolution of code clones

研究代表者

吉田 則裕 (Yoshida, Norihiro)

名古屋大学・情報学研究科・准教授

研究者番号：00582545

交付決定額(研究期間全体)：(直接経費) 1,900,000円

研究成果の概要(和文)：コードクローンのリファクタリングとは、ソースコード中に含まれるコードクローンの集合(互いに類似したコード片の集合)を単一の関数にまとめることである。あるコード片を修正するとその全てのコードクローンを見つけ出し、修正を検討する必要があるため、コードクローンをリファクタリングすることで保守コストを削減できる。本研究では、統合開発環境を使用する開発者の編集作業をオンライン分析することで、コードクローンに対する変更に対応するリファクタリングを支援する環境を構築した。

研究成果の学術的意義や社会的意義

開発者の作業内容に応じて活用支援を行う研究は少なく、リリース版のソースコードを対象としてコードクローン検出を行い、そのフィルタリングの精度向上を目的としたものが多い。版管理システムとの連携を行う手法はいくつか存在するが、本研究のように作業内容を分析することで作業内容に応じた支援を行う研究は、コードクローンに関する研究において申請者が知る限り存在しない。今後、本研究が発展することにより、作業内容の分析に基づいて、検出したコードクローンを活用するツールの開発が盛んになると期待される。

研究成果の概要(英文)：Clone refactoring is a process of merging a set of code clones in source code into a single function. Once a developer modifies one of code clones, he/she has to determine whether or not to modify each of the others. Therefore, clone refactoring decreases the cost of software maintenance. In this research, I have proposed a refactoring environment for the evolution of code clones. The proposed environment monitors code modifications by developers on the fly.

研究分野：ソフトウェア工学

キーワード：コードクローン リファクタリング

1. 研究開始当初の背景

コードクローンに対する開発コストを削減する方法の 1 つとして、コードクローンのリファクタリングが挙げられる。コードクローンのリファクタリングとは、コードクローンの集合(互いに類似したコード片の集合)を単一の関数にまとめることである。コードクローンのリファクタリングを行うことで、コードクローンの修正にかかるコストを予防することができる。これまでに研究代表者を含む研究者は、ソースコード中からリファクタリング対象のコードクローンを自動抽出することで、コードクローンを対象としたリファクタリング支援を行う手法を提案してきている。しかし、リファクタリング支援手法には、優れたリファクタリング候補を提示できるだけでなく、開発者の作業内容に応じて適切な時期に支援を行うことが求められる。その理由は、開発者が作業内容に関連したリファクタリングを推薦すると、作業内容に関する記憶を参照しながら効率的にリファクタリングできるからである。申請者が以前共同で研究を行った実務者から、「テストが完了し、別作業を開始した段階でリファクタリング推薦しても、変更作業に関する記憶が曖昧になっており、かつ変更内容の妥当性を確認するためのテストをやり直す必要があるため、効率的なリファクタリングにかかるコストが大きくなってしまふ」という指摘があった。

2. 研究の目的

本研究では、開発者の作業内容をモニタリング・分析し、その結果に基づいてコードクローンのリファクタリングを支援する環境を構築することを目的とした。本環境では、以下の支援を行うことを目指した。

- コードクローンの作りこみをモニタリングし、作りこんだコードクローンがリファクタリング候補であれば、開発者へ通知する。作りこんだコードクローン全てがリファクタリング対象ではないため、リファクタリング候補であるもののみを通知する。
- 開発者のリファクタリング作業をモニタリングすることで、現在のリファクタリングを行っているコード片のコードクローンを検知し、あわせてリファクタリングを検討するよう開発者に推薦する。

3. 研究の方法

平成 28 年度は、本研究で提案する「開発者の作業内容をモニタリング・分析し、その結果に基づいてコードクローンのリファクタリングを支援する環境」を実装した。具体的には、まず統合開発環境上で編集中のソースコードからコードクローンを検出しリファクタリング候補として提示するツールを開発した。次に、開発者が行うリファクタリング作業を検知し、作業対象のコード片のコードクローンに対して、同種のリファクタリングを推薦するプラグインを開発した。平成 29 年度は、開発したプラグインの評価と改善を行った。評価では、大学院生を対象とした予備実験を行い、その後企業の開発者を対象とした実験を行うことで、改善点の洗い出しを行った。

4. 研究成果

本研究では、統合開発環境を使用する開発者の編集作業をオンライン分析することで、コードクローンに対する変更に対応するリファクタリングを支援する環境を構築した。本環境の実現により、開発者はコードクローンに対する変更を検知し、適切なリファクタリングを行うことで、保守性の高いソースコードを開発することができる。

5. 主な発表論文等

[雑誌論文](計 3 件)

石津 卓也, 吉田 則裕, 崔 恩瀨, 井上 克郎: "コードクローンのリファクタリング可能性に基づいた削減可能ソースコード量の分析", 情報処理学会論文誌, 査読有, Vol.60, No.4, pp.1051-1062, 2019 年。

横井 一輝, 崔 恩瀨, 吉田 則裕, 井上 克郎: "情報検索技術に基づく細粒度ブロッククローン検出", コンピュータソフトウェア, 査読有, Vol.35, No.4, pp.16-36, 2018 年。
Reishi Yokomori, Norihiro Yoshida, Masami Noro, Katsuro Inoue: "Changes of

Evaluation Values on Component Rank Model by Taking Code Clones into Consideration", IEICE Transactions on Information and Systems, 査読有, Vol.E101-D, No.1, pp.130-141, 2018.

DOI: <https://doi.org/10.1587/transinf.2017EDP7125>

[学会発表](計 31 件)

Norihiro Yoshida, Seiya Numata, Eunjong Choi, Katsuro Inoue: "Proactive Clone Recommendation System for Extract Method Refactoring", 3rd International Workshop on Refactoring (IWor 2019), 2019.

Hirota Honda, Shogo Tokui, Kazuki Yokoi, Eunjong Choi, Norihiro Yoshida, Katsuro Inoue: "CCEvovis: A Clone Evolution Visualization System for Software Maintenance", IEEE/ACM 27th International Conference on Program Comprehension (ICPC 2019), 2019.

本田 紘貴, 徳井 翔梧, 横井 一輝, 崔 恩瀨, 吉田 則裕, 井上 克郎: "コードクローン保守支援を目的とした変更履歴可視化システム", 電子情報通信学会 ソフトウェアサイエンス研究会, 2019 年.

Yuji Fujiwara, Norihiro Yoshida, Eunjong Choi, Katsuro Inoue: "Code-to-Code Search Based on Deep Neural Network and Code Mutation", 13th International Workshop on Software Clones (IWSC 2019), 2019.

Yuichi Semura, Norihiro Yoshida, Eunjong Choi, Katsuro Inoue: "Multilingual Detection of Code Clones Using ANTLR Grammar Definitions", 25th Asia-Pacific Software Engineering Conference (APSEC 2018), 2018.

Eunjong Choi, Daiki Tanaka, Norihiro Yoshida, Kenji Fujiwara, Daniel Port, Hajimu Iida: "An Investigation of the Relationship Between Extract Method and Change Metrics: A Case Study of JEdit", 25th Asia-Pacific Software Engineering Conference (APSEC 2018), 2018.

Kazuki Yokoi, Eunjong Choi, Norihiro Yoshida, Katsuro Inoue: "Investigating Vector-based Detection of Code Clones Using BigCloneBench", 25th Asia-Pacific Software Engineering Conference (APSEC 2018), 2018.

藤原 裕士, 崔 恩瀨, 吉田 則裕, 井上 克郎: "順伝播型ニューラルネットワークを用いた類似コードブロック検索の試み", 情報処理学会 ソフトウェアエンジニアリングシンポジウム 2018, 2018 年.

横井 一輝, 崔 恩瀨, 吉田 則裕, 井上 克郎: "コード片のベクトル表現に基づく大規模コードクローン集合の特徴調査", 情報処理学会 ソフトウェアエンジニアリングシンポジウム 2018, 2018 年.

徳井 翔梧, 吉田 則裕, 崔 恩瀨, 井上 克郎: "コードクローン検出が用いる局所性鋭敏型ハッシュに与えるパラメータ決定手法", 日本ソフトウェア科学会 第 35 回大会, 2018 年.

瀬村 雄一, 吉田 則裕, 崔 恩瀨, 井上 克郎: "構文定義記述を用いた多言語対応コードクローン検出ツールの開発", 情報処理学会 第 199 回ソフトウェア工学研究発表会, 2018 年.

Norihiro Yoshida, Takuya Ishizu, Buford Edwards III, Katsuro Inoue: "How Slim Will My System Be? Estimating Refactored Code Size by Merging Clones", IEEE/ACM 26th International Conference on Program Comprehension (ICPC 2018), 2018.

徳井 翔梧, 吉田 則裕, 崔 恩瀨, 井上 克郎: "局所性鋭敏型ハッシュを用いたコードクローン検出のためのパラメータ決定手法", 電子情報通信学会 ソフトウェアサイエンス研究会, 2018 年.

Yuichi Semura, Norihiro Yoshida, Eunjong Choi, Katsuro Inoue: "CCFinderSW: Clone Detection Tool with Flexible Multilingual Tokenization", 24th Asia-Pacific Software Engineering Conference (APSEC 2017), 2017.

沼田 聖也, 吉田 則裕, 崔 恩瀨, 井上 克郎: "開発作業のモニタリングによるコードクローン集約支援環境の構築", 情報処理学会 第 197 回ソフトウェア工学研究発表会, 2017 年.

山田 悠貴, 崔 恩瀨, 吉田 則裕, 飯田 元: "凝集度を用いたメソッドのインライン化の支援手法", 情報処理学会 第 197 回ソフトウェア工学研究発表会, 2017 年.

石津 卓也, 吉田 則裕, 崔 恩瀨, 井上 克郎: "コードクローンに対するリファクタリング可能性に基づいた削減可能ソースコード量の調査", 情報処理学会 第 197 回ソフトウェア工学研究発表会, 2017 年.

横井 一輝, 崔 恩瀨, 吉田 則裕, 井上 克郎: "情報検索技術に基づくブロッククローン検出", 情報処理学会 第 196 回ソフトウェア工学研究発表会, 2017 年.

様式 C - 19、F - 19 - 1、Z - 19、CK - 19 (共通)

瀬村 雄一, 吉田 則裕, 崔 恩瀨, 井上 克郎: "多言語対応のための字句解析機構を持つコードクローン検出ツールの開発", 情報処理学会 第 196 回ソフトウェア工学研究発表会, 2017 年.

田中 大樹, 崔 恩瀨, 吉田 則裕, 藤原 賢二, 飯田 元: "プロセスメトリクスを用いたメソッド抽出事例の調査と予測モデルの構築", 電子情報通信学会ソフトウェアサイエンス研究会, 2017 年.

- ⑳ Katsuhisa Maruyama, Shinpei Hayashi, Norihiro Yoshida, Eunjong Choi: "Frame-Based Behavior Preservation in Refactoring", 24th IEEE International Conference on Software Analysis, Evolution, and Reengineering (SANER 2017), 2017.
- ㉑ Norihiro Yoshida: "When, why and for whom do practitioners detect technical debts?: An experience report", 1st International Workshop on Technical Debt Analytics (TDA 2016), 2016.
- ㉒ 上村 恭平, 吉田 則裕, 崔 恩瀨, 飯田 元, 曲 生国, 秋庭 真一: "ソースコードの削減可能量計測ツールの開発", 日本ソフトウェア科学会 第 23 回ソフトウェア工学の基礎ワークショップ (FOSE 2016), 2016 年.
- ㉓ 中村 勇太, 崔 恩瀨, 吉田 則裕, 春名 修介, 井上 克郎: "複数プログラミング言語で記述されたソフトウェアからのコードクローン検出", 情報処理学会 第 194 回ソフトウェア工学研究発表会, 2016 年.
- ㉔ Seiya Numata, Norihiro Yoshida, Eunjong Choi, Katsuro Inoue: "On the Effectiveness of Vector-based Approach for Supporting Simultaneous Editing of Software Clones", 17th International Conference on Product-Focused Software Process Improvement (PROFES 2016), 2016.
- ㉕ 田中 大樹, 吉田 則裕, 藤原 賢二, 崔 恩瀨, 飯田 元: "プロセスメトリクスを用いたメソッド抽出事例の特徴調査", ソフトウェアエンジニアリングシンポジウム 2016, 2016 年.
- ㉖ Shogo Tsutsumi, Eunjong Choi, Norihiro Yoshida, Katsuro Inoue: "Graph-Based Approach for Detecting Impure Refactoring from Version Commits", 1st International Workshop on Software Refactoring (IWor 2016), 2016.
- ㉗ 沼田 聖也, 吉田 則裕, 崔 恩瀨, 井上 克郎: "欠陥の同時修正支援における関数クローン検出ツールの有効性調査", 電子情報通信学会ソフトウェアサイエンス研究会, 2016 年.
- ㉘ 石津 卓也, 吉田 則裕, 崔 恩瀨, 井上 克郎: "メタヒューリスティクスを用いた集約可能コードクローン量の推定", 情報処理学会 第 193 回ソフトウェア工学研究発表会, 2016 年.
- ㉙ 堤 祥吾, 吉田 則裕, 崔 恩瀨, 井上 克郎: "探索的手法による impure リファクタリング検出", 情報処理学会 第 193 回ソフトウェア工学研究発表会, 2016 年.
- ㉚ Norihiro Yoshida, Tsubasa Saika, Eunjong Choi, Ali Ouni, Katsuro Inoue: "Revisiting the Relationship Between Code Smells and Refactoring", 24th IEEE International Conference on Program Comprehension (ICPC 2016), 2016.

〔産業財産権〕

出願状況 (計 0 件)

取得状況 (計 0 件)

〔その他〕

ホームページ等

CCFinderSW <https://github.com/YuichiSemura/CCFinderSW>

6. 研究組織

研究分担者は存在しない。

科研費による研究は、研究者の自覚と責任において実施するものです。そのため、研究の実施や研究成果の公表等については、国の要請等に基づくものではなく、その研究成果に関する見解や責任は、研究者個人に帰属されます。