

機関番号：13901

研究種目：若手研究(B)

研究期間：2009～2010

課題番号：21700027

研究課題名(和文)

デバッグ履歴およびコードスニペットを対象としたソフトウェア開発経験マイニング

研究課題名(英文)

Mining Development Experiences from code snippets and interaction history on debugging

研究代表者：

小林 隆志 (Takashi Kobayashi)

名古屋大学・情報科学研究科・准教授

研究者番号：50345386

研究成果の概要(和文)：

本研究では、ソフトウェア開発者がライブラリや再利用部品を利用して試行錯誤しながら実装する際の、開発活動、実行履歴および結果として残る利用のための実装コード群を解析する手法を開発し、開発経験をマイニングするいくつかの手法を提案した。開発履歴に対するマイニングでは、ファイル参照といった詳細な活動履歴を用いることで精度の高い変更支援を実現した。また、ソースコードに対するマイニング手法を検討し、JavaScriptのブラウザ互換問題のための代替コード発見手法などを提案した。

研究成果の概要(英文)：

We have developed several methods to analyze source codes, execution traces and history of developer's activities and to extract development experience from them when developers use reusable components. We have shown that our proposal method based on a developer's activity mining can improve accuracy of change guide. We also proposed several mining applications for source codes such that a method to find alternative codes of JavaScript for browser compatibility problem.

交付決定額

(金額単位：円)

	直接経費	間接経費	合計
2009年度	1,500,000	450,000	1,950,000
2010年度	1,600,000	480,000	2,080,000
総計	3,100,000	930,000	4,030,000

研究分野：ソフトウェア工学

科研費の分科・細目：ソフトウェア

キーワード：ソフトウェア保守，変更支援，ソフトウェア再利用，データマイニング

1. 研究開始当初の背景

ソフトウェア開発工程では、多数の設計・実装上の問題に対し適切な選択を行うことが重要である。その際、類似の問題に対する他者の問題解決経験を参照することは直面する問題の解決方法を検討する上で非常に有益である。このような“経験の再利用”は、分析、設計工程では、デザインパターン、実装工程においては、言語毎のイデオムやプログラミングスタイルといった形で、熟練

の開発者の経験を個人や小規模な組織が文書化せずに共有している暗黙知の状態から、より多くの開発者と共有するための形式知へと変換する試みがなされてきた。

これに加え、近年では、再利用可能な部品群も、非常に完成度の高いものがWWW上に公開されるようになってきた。企業内においてもプロダクトライン開発におけるコア資産として再利用部品の管理を行う動きが

活発である。

このように一部の開発経験やその成果物は蓄積され部分的に再利用がなされるようになってきたが、依然として熟練者の経験は十分再利用されておらず問題が残っている。

2. 研究の目的

本研究では、再利用部品を利用する際の実装工程には、これまで形式化されてこなかった再利用可能な経験が存在することに着眼する。ソフトウェア開発者がライブラリや再利用部品を利用して試行錯誤しながら実装する際の、デバッグ履歴および結果として残る利用のための実装コード群から、その開発経験をマイニングし形式化することで、多くの開発者を支援する手法を開発することを目的とする。

実装の最終結果であるソースコードだけでなく、その試行錯誤過程である、デバッグ履歴をも利用することで、失敗原因とその解決策のコード片を含む高品質の経験を抽出することが可能となり、また、人手による形式化ではなく、多種多様なソフトウェアリポジトリを情報源としマイニングすることで組織や開発ドメインに偏る事の無い広い範囲の経験を効果的に収集し、支援に利用することが可能となる。

3. 研究の方法

本研究は、1) 開発活動履歴の蓄積およびそれに対するマイニング、2) 多量の実行履歴の効果的な可視化による理解支援およびマイニング、3) ソースコードに対するマイニングの3種類のマイニング応用によりソフトウェア開発経験を抽出することを目指す。

特に、開発活動履歴においては、ソースコードの変更だけでなく、その変更に至るまでに参照したファイルや、コーディング規約違反などツールにより観測可能な特性を詳細に蓄積活用することで、後の開発に行かせる知識を抽出する。

4. 研究成果

本研究課題の成果は大きく分けて以下の5つである。

1) 作業履歴蓄積ツールを応用した保守作業支援手法の開発

本研究では、まず、試行錯誤の作業履歴をより詳細に保存するために、デバッグ時の例外発生情報に加え、テストコードおよびその実行履歴を収集するツールを開発し、それらを

対象としたマイニング手法を試作した。

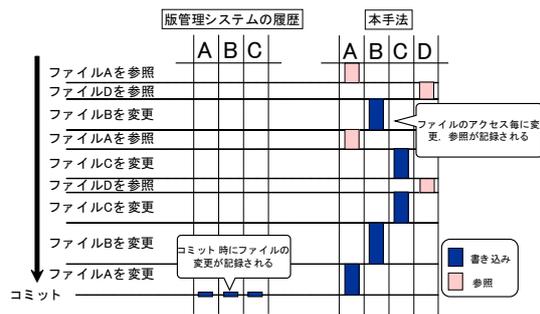
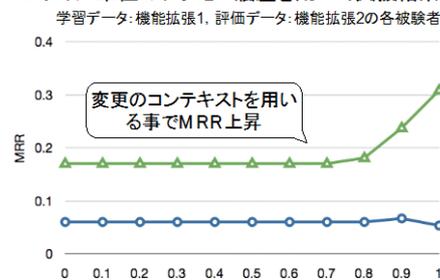


図:提案手法で保存される履歴情報

さらに、その統合開発環境における開発者の作業履歴を蓄積するツールを用い、保守活動において困難とされる波及解析問題に対して、変更伝播を予測し次に修正が必要となる箇所を推薦する手法を提案した。

ファイル単位のアクセス履歴を用いて支援結果



α の値が高くなるにつれてMRRが増加
=変更支援で変更のコンテキストを用いる事が有効

図:実験結果

提案手法では、ファイル参照などの保守活動の詳細な作業ログをマイニングすることで予測モデルを作成し、参照回数や時間をコンテキスト情報として用いることで精度の高い推薦を実現した。

2) コーディングルールチェッカーの開発と、その応用

開発活動の蓄積をする対象として、コーディングルールチェッカーを検討し、C言語を対象とするコーディングルールチェッカーの開発を行った。開発環境によって個別に設定されるコーディング規約に柔軟に対応できるように、カスタマイズ性が高く、ルールの追加が容易となるように設計し、実際に代表的なコーディング規約に適用できることを確認した。

さらに、開発活動情報としてコーディング規約違反に着目し、プロジェクトの状態や開発者を評価するためのソフトウェアメトリ

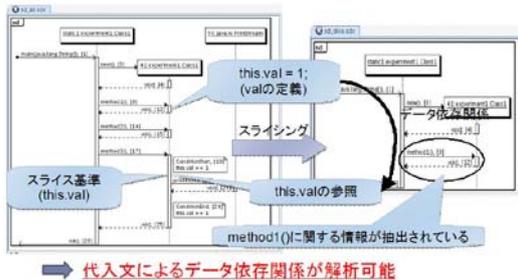
クスを提案した。提案手法ではコーディングルールチェッカーの出力結果を計測し、ルールや開発者毎の出現頻度、時間分布、空間分布を用いる複数のメトリクスを提案し、オープンソースソフトウェアの開発履歴を用いてその効果を測定する実験を行った。



図：カスタマイズ性に優れたコーディングチェッカー CX-Checker

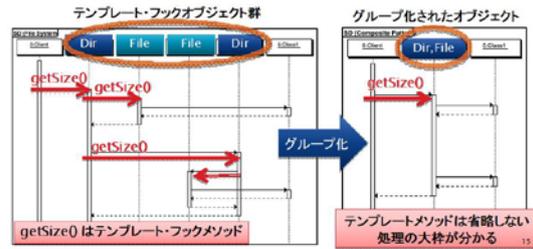
3) 実行履歴の解析と効果的なデバッグ支援手法の開発

プログラムを実行した際の実行履歴は膨大な情報となり、全てを把握することが困難である。本研究ではまず、Java プログラムをデバッグする際に、実行履歴を効果的に可視化するための手法として Sequence Diagram Slicing 手法を開発した。提案手法を用いることで目標の振る舞いに関連する部分のみの理解しやすいシーケンス図を作成することが可能となる。本手法を応用することで、試行錯誤における実行履歴やそのデバッグ履歴を効果的に可視化し形式知化のための分析を行うことが可能となる。



図：シーケンス図のスライシング結果

さらに、より効果的に可視化するための手法としてメタパターンに基づく抽象化手法も提案した。提案手法によって、振る舞い理解の観点では不要となる相互作用をカプセル化し、可視化対象を大幅に削減することを可能とした。前述の slicing 手法と併用することで、大規模な実行履歴に対しても着目箇所を効果的に可視化することが可能となる。

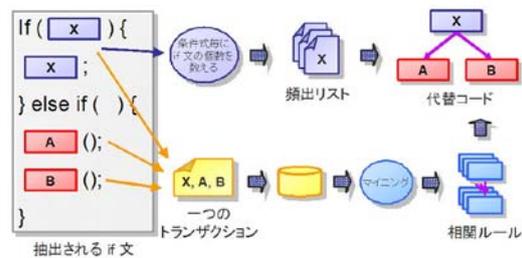


図：メタパターンを用いたグループ化

また、上記の解析ツールを応用して、実行履歴に対してマイニングすることで、通常とは異なる振る舞いを検知し、不具合箇所の候補を提示するデバッグ支援手法の開発も行った。提案手法により限定的ではあるが、発見することが困難な不具合箇所を効果的に発見できることを確認した。

4) コーディングパターンのマイニング

まず、Java 言語に対して、構文木の特徴から類似するソースコードを発見し、利用テンプレートをマイニングする手法を検討した。さらに、メソッド呼び出しの内容をインライン展開することにより、複数のメソッドに分断されているテンプレートを発見するための手法を検討し、予備実験を行った。さらに、クライアントサイド言語として広く使われる JavaScript のためのマイニング手法を開発した。



図：提案手法の概要

JavaScript は実行環境が多様であるという特徴があり、実行される環境を考慮した実装が必要となる。本研究では、Web 上に散在する JavaScript コードを収集し、環境依存を吸収するための代替コードを発見する手法を提案した。Web 上にある実際の JavaScript コードを収集し、マイニングすることにより、代替コードが発見できることを確認した。

5) 様々な開発環境の検討

多品種開発や、MVC アーキテクチャに特化した開発手法における効果的な開発方法とその支援環境を提案しツール化と実験を行った。これらの経験をもとに、様々な開発環境における 試行錯誤情報を効果的に取り扱う

方式の検討を行った。

5. 主な発表論文等

(研究代表者、研究分担者及び連携研究者には下線)

[雑誌論文] (計 3 件)

- (1) 蛭牟田英治, 小林隆志, 末次亮, 大須賀俊憲, 山本晋一郎, 阿草清滋, JavaScript ソースコードリポジトリを用いた代替コード発見, コンピュータソフトウェア, 27(2), 129-134, 2010(査読有)
- (2) 上原伸介, 小林隆志, 大須賀俊憲, 山本晋一郎, 阿草清滋, 多粒度な可視化によるソフトウェア理解支援, コンピュータソフトウェア, 27(2), 116-121, 2010(査読有)
- (3) 小林隆志, 林晋平, データマイニング技術を応用したソフトウェア構築・保守支援の研究動向, コンピュータソフトウェア, 27(3), 13-23, 2010 (査読有)

[学会発表] (計 22 件)

- (1) 野田訓広, 小林隆志, 阿草清滋, メタパターン適用情報に基づくオブジェクトの協調動作履歴可視化ツール, 情報処理学会ソフトウェア工学研究会, 2011.03.15, 東京
- (2) 加藤希, 小林隆志, 阿草清滋: 変更支援のための成果物アクセス履歴マイニング, 電子情報通信学会 ソフトウェアサイエンス研究会, 2011.03.07, 沖縄
- (3) 秋山幹博, 林晋平, 小林隆志, 佐伯元司: 責務記述に基づくクラスの責務割り当て支援, 電子情報通信学会 ソフトウェアサイエンス研究会, 2011.03.07, 沖縄
- (4) 池田健人, 小林隆志, 波多野賢治, 深川大路: 複数観点をを用いた木類似度算出法によるソースコードの類似構造の抽出 第3回データ工学と情報マネジメントに関するフォーラム (DEIM2011), 2011.2.28, 静岡
- (5) 高井康勢, 渥美紀寿, 小林隆志: コーディング規約違反に着目したソフトウェアメトリクス, IPSJ ウィンターワークショップ 2011 イン修善寺, 2011.1.21, 静岡
- (6) 上原伸介, 小林隆志, 渥美紀寿, 山本晋一郎, 阿草清滋: トレースに基づくデバッグにおける欠陥箇所発見支援手法, 第17回 ソフトウェア工学の基礎ワークショップ FOSE2010 2010.11.17, 新潟
- (7) 渥美紀寿, 谷聡貴, 大塚直也, 小林隆

志, 山本晋一郎, 阿草清滋: 組込みソフトウェア製品ファミリにおけるコンパイルスイッチによる構成管理手法, 第17回 ソフトウェア工学の基礎ワークショップ FOSE2010, 2010.11.17, 新潟

- (8) 小林隆志, 林晋平: データマイニング技術を応用したソフトウェア構築・保守支援, 第17回 ソフトウェア工学の基礎ワークショップ FOSE2010, 2010.11.17, 新潟
- (9) 渥美紀寿, 山本晋一郎, 小林隆志, 阿草清滋: CASE ツール・プラットフォームのための C ソースプログラムの XML 表現とその応用. ソフトウェアエンジニアリングシンポジウム 2010, 2010.9.1. 東京
- (10) Tetsutaro Watanabe, Takashi Kobayashi, Haruo Yokota: Searching Keyword-lacking Files Based on Latent Interfile Relationships, 5th International Conference on Software and Data Technologies (ICSOF2010), 2010.7.22, Athens Greece
- (11) 横山祐司, 日高隆博, 山本晋一郎, 小林隆志, 手嶋茂晴, 阿草清滋: バリエーション並行開発のための版管理ツールと統合開発環境, 情報処理学会 第167回 ソフトウェア工学研究会, 2010.3.17, 東京
- (12) 林英志, 日高隆博, 山本晋一郎, 小林隆志, 上原正太, 間瀬順一, 鈴木延保, 阿草清滋: メタ情報とコンテキスト情報を用いた入力補完機能と XPath 入力への応用, 情報処理学会 第167回 ソフトウェア工学研究会, 2010.3.17, 東京
- (13) 関文貴, 日高隆博, 山本晋一郎, 小林隆志, 手嶋茂晴, 阿草清滋: 制御ソフトウェアの固定小数点演算化ツールの設計と実装 情報処理学会 第167回 ソフトウェア工学研究会, 2010.3.17, 東京
- (14) 小林隆志, 大須賀俊憲, 上原伸介, 蛭牟田英治, 林英志, 間瀬順一, 山本晋一郎, 渥美紀寿, 川口直弘, 鈴木延保, 阿草清滋: CX-Checker: 柔軟なカスタマイズが可能な C 言語コーディングルールチェッカー, 情報処理学会全国大会, 2010.3.4 東京
- (15) Eiji Hirumuta, Takashi Kobayashi, Noritoshi Atsumi, Kiyoshi Agusa, Finding Alternate Javascript Codes with a Code Repository, the IASTED International Conference on Software Engineering 2010.2.17,

- Innsbruck, Austria
- (16) 小林隆志: 再利用支援情報発見のためのプログラム解析技術, ウィンターワークショップ 2010・イン・倉敷, 2010.1.25, 岡山
 - (17) Kunihiro Noda, Takashi Kobayashi, Shinichiro Yamamoto and Kiyoshi Agusa: Sequence Diagram Slicing, 16th Asia-Pacific Software Engineering Conference (APSEC2009), 2009.12.1 Penang, Malaysia
 - (18) 蛭牟田英治, 末次亮, 大須賀俊憲, 小林隆志, 山本晋一郎, 阿草清滋: JavaScript ソースコードリポジトリを用いた代替コード発見, 第16回ソフトウェア工学の基礎ワークショップ FOSE 2009, 2009.11.13, 箱根
 - (19) Hiroshi Kazato, Rafael Weis, Shinpei Hayashi, Takashi Kobayashi, Motoshi Saeki: Model-View-Controller Architecture Specific Model Transformation, Proc. 9th OOPSLA Workshop on Domain-Specific Modeling (DSM09), 2009.10.25, Orland, USA
 - (20) 大須賀俊憲, 小林隆志, 間瀬順一, 渥美紀寿, 山本晋一郎, 鈴木延保, 阿草清滋: CX-Checker: C言語プログラムのためのカスタマイズ可能なコーディングチェッカ, ソフトウェアエンジニアリングシンポジウム (SES2009) 2009.9.7, 東京
 - (21) 大場光明, 大須賀俊憲, 小林隆志, 山本晋一郎: メソッドのインライン展開によるオブジェクトの責務単位でのコードスニペットの抽出, ソースコードの類似性ワークショップ, 2009.9.7, 東京
 - (22) Chao-Qing Lv, Takashi Kobayashi, Kiyoshi Agusa, Wu Kun, Zhu Qing: Image Semantic Search Engine, Proc. Intl Workshop on Database Technology and Applications (DBTA2009), 2009.4.22, 武漢, 中国

6. 研究組織

(1) 研究代表者

小林 隆志 (KOBAYASHI Takashi)
名古屋大学・情報科学研究科・准教授
研究者番号: 50345386

(2) 研究分担者

なし

(3) 連携研究者

なし