

令和 6 年 5 月 24 日現在

機関番号：14401

研究種目：若手研究

研究期間：2019～2023

課題番号：19K20239

研究課題名（和文）Web上のAPI利用例に対する情報の鮮度を判定する整合性検査手法の開発

研究課題名（英文）Development of an integrity checking method to determine the freshness of API examples on the Web

研究代表者

神田 哲也（KANDA, Tetsuya）

大阪大学・大学院情報科学研究科・助教

研究者番号：90780726

交付決定額（研究期間全体）：（直接経費） 2,800,000円

研究成果の概要（和文）：研究期間全体を通じ、以下の成果を得た。ライブラリのリポジトリから、整合性違反検出のための解析基盤となるデータを抽出する技術を開発した。そのうえで、Java言語、Python言語のコード片に対し、どのバージョンなら動作するかという情報の鮮度を判定する整合性違反を検出する手法を開発した。Python言語についてはブラウザ拡張の形でツールを実装し、またその必要性をQ&Aサイト上のコード片の分析を通じて示した。より鮮度の高い情報へ誘導するための仕組みを実現するには至らなかったが、そのための基盤技術としてコード片のメトリクスによる分類技術や類似度計算技術の改善、動的な実行の記録手法の開発と評価を行った。

研究成果の学術的意義や社会的意義

本研究で扱ったWeb上の情報について、その鮮度が重要であることを示し、またその検出のための技術開発を行うことができた。ソフトウェア自体が古くなり動作しなくなることはソフトウェア工学よく研究されているが、その概念を他のQ&Aサイトなどに広げることが可能でありまた重要であることを示した。また、Web上の情報は研究者に限らずプログラミングを行う人すべてがよく参照するものであるから、そのような情報をより正確に利活用するための概念を提唱し、実装したことは社会的意義が大きいと考える。

研究成果の概要（英文）：In this research, we got the following achievements.

We developed a technique for extracting data from library repositories that can be used as the basis of analysis for consistency violation detection. We developed a method to determine which versions of libraries are used in Java code snippets. For Python, we implemented a tool to detect which versions of Python are used in the code fragments on a Q&A site, implemented a browser extension, and demonstrated its necessity.

Although we were not able to develop a mechanism for guiding users to more fresh information, we developed a method for classifying code fragments based on metrics, improving similarity calculation techniques, and recording dynamic execution as fundamental technologies for this research project.

研究分野：ソフトウェア工学

キーワード：ソフトウェア工学 データマイニング API ソフトウェア開発効率化 整合性

1. 研究開始当初の背景

ソフトウェア開発の際に、開発者は様々なドキュメントを参照しながらプログラムを書いていく。ライブラリの公式ドキュメントや技術書、個人ブログなどの Web ページ、またソフトウェア開発に特化した Q&A サイトなど、多様なドキュメントが存在する。特に Web 上の情報は、世界中の開発者から日々新たな知識が発信・蓄積されており、またそれを即座に参照することができる。

その一方、ソフトウェア開発に関わる環境も日々進歩している。脆弱性への対応や、新機能の追加などの理由により、ライブラリ自体も頻繁に更新される。ライブラリの仕様が変更されたことによって、例えばそれまで Q&A サイトに記載されていたコードをコピーすれば動作していたものが、そのままではエラーとなることも考えられる。しかし、そのコードがライブラリのどの時点の版に対応していたのか、またその後の仕様変更の影響を受けているのかは、ライブラリの仕様に詳しくなければすぐには判別できない。

また、回答後に API に変更が起こり、そのコード例が利用不可能になることも考えられる。Q&A サイトでは質問や回答に投票をすることが可能であるが、過去の高評価は、回答の現在の API との整合性を保証しない。このような状況を踏まえると、Web 上の API 利用例に関する情報が今でも正確であることを保証するための仕組みが必要であると考えた。

2. 研究の目的

本研究では、Web 上で手に入るソフトウェア開発に関する API 利用例について、その利用例が最新版の API に対応したものであるかどうか、その情報の鮮度を判定する整合性検査手法を開発する。また、それをブラウザ拡張として社会に提供する。

3. 研究の方法

本研究は以下の 3 つの段階に分けて研究を行った。

(1) 解析基盤の準備

ライブラリおよびその開発履歴を解析し、API 呼び出しに関する情報を抽出する。整合性違反検出のために、その情報を格納した DB を作成する。ライブラリの過去のバージョンを含めた解析を行う。

(2) 整合性違反検出手法の開発

Q&A サイト上のコード片について、API 呼び出し部分を抽出する。一般的に Q&A サイト上のコード片はプログラムの一部分だけを記述していることが多いため、部分的なコードであっても内部で利用されているライブラリや API 呼び出しを特定するための工夫が必要である。その後、DB との照合により、どのライブラリのどのバージョンがコード片において想定されているのかを確認する。

(3) 整合性違反への対処の推薦

違反の内容をもとに、開発者へ対処を促す仕組みの構築を考える。しかし、変更された API の最新版に関する投稿が常に存在するとは限らない。そのため、まずは関連しそうな API の変更を提示することを主な目標とする。

4. 研究成果

研究期間全体を通じ、以下の成果を得た。

(1) 解析基盤の準備

Java ライブラリのリポジトリから、整合性違反検出のための解析基盤となる API 呼び出しのデータを抽出する技術を開発した。また、ライブラリを利用するアプリケーション側のリポジトリを分析することにより、API そのものだけでなく、API の組み合わせパターンも変化していくことがわかった。主流なパターンは長く多く使われることや、パターンが複数プロジェクトで同時に減少する現象などを確認した。

また、上記の研究は静的な解析情報を用いていたが、API の変更に対するプログラムの挙動を動的に観測するための技術開発を行い、Java ライブラリのテストケースの実行について新旧で比較し可視化する手法を提案した。

(2) 整合性違反検出手法の開発

Java のライブラリについて、Q&A サイトの投稿を分析し DB と照合するツールの試作を行った。

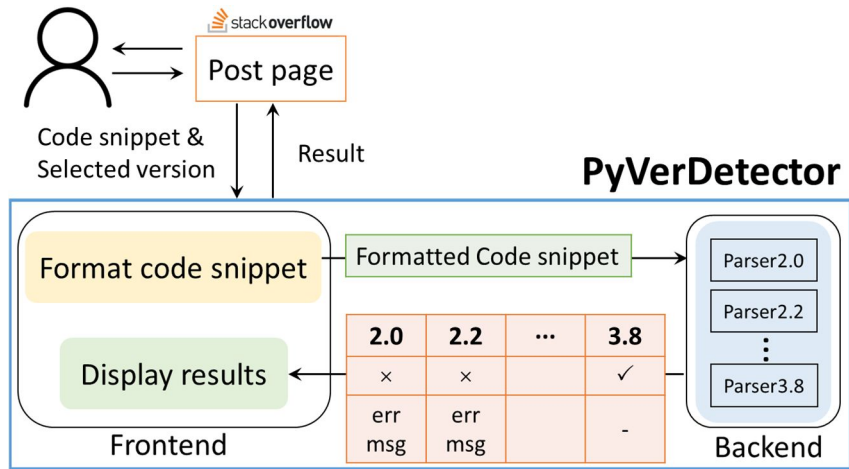


図 1: Q&A サイト上の Python 言語で書かれたコード片に対し、整合性違反を通知するブラウザ拡張の全体図

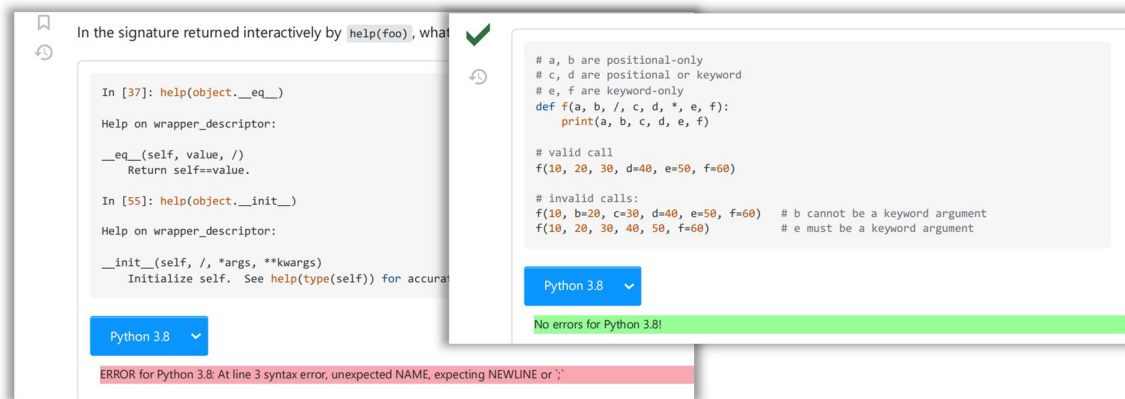


図 2: 整合性違反を検出した場合 (左) と違反がなかった場合 (右) のブラウザ上での表示

いくつかの Java ライブラリについて整合性違反となる投稿を発見できた。

また、Python 言語のコード片に対し、Q&A サイトの投稿に含まれるコード片について言語がどのバージョンなら動作するかという情報の鮮度を判定する整合性違反を検出する手法を開発した(図 1、図 2)。ブラウザ拡張の形でツールを実装し、またその必要性を Q&A サイト上のコード片の分析を通じて示した。

(3) 整合性違反への対処の推薦

より鮮度の高い情報へ誘導するための仕組みは、鮮度の高い情報を確実に特定する手法に至らず、完成しなかったが、上記のツールのように動作する Python 言語のバージョンを表示するなど、整合性違反に関する情報をユーザーに提示することで対処を促すための研究を進めた。

API 利用例が変更された際に、単にその情報の鮮度を判定するだけでなく、それを利用している側のソフトウェアがその変更に従うべきかを調査した。実際のソフトウェアを調査し、API 利用例の変更にソフトウェア側が追従していないケースについて分類を行った。その結果、ソフトウェア保守の観点から、いくつかの種類の変更において追従の必要性が高い、もしくはその必要性を検討するべきであるといえるものがあることを確認した。

整合性違反検出手法の実利用シーンへの適用に向けて、エディタに組み込んで開発中のソフトウェアの変更を逐一記録できる細粒度編集履歴収集プラットフォームを試作した。

5. 主な発表論文等

〔雑誌論文〕 計8件（うち査読付論文 8件/うち国際共著 1件/うちオープンアクセス 2件）

1. 著者名 Shiyu Yang, Tetsuya Kanda, Daniel M German, Yoshiki Higo	4. 巻 E107-D
2. 論文標題 Unveiling Python Version Compatibility Challenges in Code Snippets on Stack Overflow	5. 発行年 2024年
3. 雑誌名 IEICE TRANSACTIONS on Information and Systems	6. 最初と最後の頁 -
掲載論文のDOI（デジタルオブジェクト識別子） 10.1587/transinf.2023EDP7238	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 該当する
1. 著者名 Shimari Kazumasa, Ishio Takashi, Kanda Tetsuya, Inoue Katsuro	4. 巻 236
2. 論文標題 Evaluating the effectiveness of size-limited execution trace with near-omniscient debugging	5. 発行年 2024年
3. 雑誌名 Science of Computer Programming	6. 最初と最後の頁 -
掲載論文のDOI（デジタルオブジェクト識別子） 10.1016/j.scico.2024.103117	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -
1. 著者名 小池 耀、眞鍋 雄貴、神田 哲也、井上 克郎、肥後 芳樹	4. 巻 64
2. 論文標題 Javaを用いたプロジェクトおよびKotlinを用いたAndroidアプリケーションを対象としたビルド可能性調査	5. 発行年 2023年
3. 雑誌名 情報処理学会論文誌	6. 最初と最後の頁 1394 ~ 1398
掲載論文のDOI（デジタルオブジェクト識別子） 10.20729/00227616	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -
1. 著者名 田邊 傑士、神田 哲也、眞鍋 雄貴、井上 克郎、肥後 芳樹	4. 巻 J106-D
2. 論文標題 Debianパッケージ間の依存関係を表すSPDXドキュメント自動生成ツールの開発	5. 発行年 2023年
3. 雑誌名 電子情報通信学会論文誌D 情報・システム	6. 最初と最後の頁 457 ~ 458
掲載論文のDOI（デジタルオブジェクト識別子） 10.14923/transinfj.2023JDL8004	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -

1. 著者名 栗原 拓己, 嶋利 一真, 神田 哲也, 井上 克郎	4. 巻 J105-D
2. 論文標題 Stack Overflowのコード片へ加えられた変更に従わないGitHubプロジェクトの変更パターン分類による考察	5. 発行年 2022年
3. 雑誌名 電子情報通信学会論文誌D	6. 最初と最後の頁 717 ~ 719
掲載論文のDOI (デジタルオブジェクト識別子) 10.14923/transinfj.2022JDL8007	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -

1. 著者名 Kazumasa Shimari, Takashi Ishio, Tetsuya Kanda, Naoto Ishida, Katsuro Inoue	4. 巻 206
2. 論文標題 NOD4J: Near-Omniscient Debugging Tool for Java Using Size-Limited Execution Trace	5. 発行年 2021年
3. 雑誌名 Science of Computer Programming	6. 最初と最後の頁 -
掲載論文のDOI (デジタルオブジェクト識別子) 10.1016/j.scico.2021.102630	査読の有無 有
オープンアクセス オープンアクセスとしている (また、その予定である)	国際共著 -

1. 著者名 伊藤 薫, 石尾 隆, 神田 哲也, 井上 克郎	4. 巻 J104-D
2. 論文標題 軽量なデータ構造を利用したソフトウェア進化履歴の高速な復元手法	5. 発行年 2021年
3. 雑誌名 電子情報通信学会論文誌D	6. 最初と最後の頁 609 ~ 621
掲載論文のDOI (デジタルオブジェクト識別子) 10.14923/transinfj.2020JDP7080	査読の有無 有
オープンアクセス オープンアクセスとしている (また、その予定である)	国際共著 -

1. 著者名 伊藤 薫, 石尾 隆, 神田 哲也, 井上 克郎	4. 巻 J103-D
2. 論文標題 軽量な類似度計算によるプロジェクト間のソースファイル集合の再利用検出	5. 発行年 2020年
3. 雑誌名 電子情報通信学会論文誌D 情報・システム	6. 最初と最後の頁 542 ~ 554
掲載論文のDOI (デジタルオブジェクト識別子) 10.14923/transinfj.2019JDP7077	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -

〔学会発表〕 計26件（うち招待講演 0件 / うち国際学会 7件）

1. 発表者名 Rio Kishimoto, Tetsuya Kanda, Yuki Manabe, Katsuro Inoue, Yoshiki Higo
2. 発表標題 Osmy: A Tool for Periodic Software Vulnerability Assessment and File Integrity Verification using SPDX Documents
3. 学会等名 The 31st International Conference on Software Analysis, Evolution and Reengineering (国際学会)
4. 発表年 2024年

1. 発表者名 音田 渉, 神田 哲也, 眞鍋 雄貴, 井上 克郎, 肥後 芳樹
2. 発表標題 Stack OverflowにおけるSBOM利活用に関する質問の分析
3. 学会等名 電子情報通信学会ソフトウェアサイエンス研究会2024年3月研究会
4. 発表年 2024年

1. 発表者名 藤原 勇真, 神田 哲也, 嶋利 一真, 肥後 芳樹
2. 発表標題 行単位の依存関係を用いたテスト選択手法の提案
3. 学会等名 ソフトウェアエンジニアリングシンポジウム2023 ポスター発表（論文あり）
4. 発表年 2023年

1. 発表者名 鬼塚 仙太郎, 神田 哲也, 眞鍋 雄貴, 肥後 芳樹
2. 発表標題 Stack Overflowと言語ドキュメントの紐づけ手法の検討
3. 学会等名 電子情報通信学会ソフトウェアサイエンス研究会2023年7月研究会
4. 発表年 2023年

1. 発表者名 岸本 理央, 神田 哲也, 眞鍋 雄貴, 井上 克郎, 肥後 芳樹
2. 発表標題 SPDXドキュメントを用いた脆弱性診断とチェックサム検証を行うツール
3. 学会等名 電子情報通信学会ソフトウェアサイエンス研究会2023年7月研究会
4. 発表年 2023年

1. 発表者名 Shiyu Yang, Tetsuya Kanda, Davide Pizzolotto, Daniel M German, Yoshiki Higo
2. 発表標題 PyVerDetector: A Chrome Extension Detecting the Python Version of Stack Overflow Code Snippets
3. 学会等名 The 31st International Conference on Program Comprehension (国際学会)
4. 発表年 2023年

1. 発表者名 Sentaro Onizuka, Tetsuya Kanda, Katsuro Inoue
2. 発表標題 Comparison of Developer 's Work Efficiency between Different Editors
3. 学会等名 29th Asia-Pacific Software Engineering Conference (国際学会)
4. 発表年 2022年

1. 発表者名 田邊 傑士, 眞鍋 雄貴, 神田 哲也, 井上 克郎
2. 発表標題 Debianパッケージに対する依存関係を含むSPDXファイルの自動生成ツール
3. 学会等名 電子情報通信学会ソフトウェアサイエンス研究会2022年7月研究会
4. 発表年 2022年

1. 発表者名 藤原 勇真, 神田 哲也, 嶋利 一真, 井上 克郎
2. 発表標題 Javaプログラムを対象としたソースコードの変更量と実行トレースの変化量間の相関調査
3. 学会等名 情報処理学会第211回ソフトウェア工学研究発表会
4. 発表年 2022年

1. 発表者名 Shiyu Yang, Tetsuya Kanda, Katsuro Inoue
2. 発表標題 The Effect of Python Version Upgrades on the Compilability of Code Snippets Posted on Stack Overflow
3. 学会等名 情報処理学会第211回ソフトウェア工学研究発表会
4. 発表年 2022年

1. 発表者名 Tetsuya Kanda, Kazumasa Shimari, Katsuro Inoue
2. 発表標題 didiff: A Viewer for Comparing Changes in both Code and Execution Traces
3. 学会等名 30th IEEE/ACM International Conference on Program Comprehension (国際学会)
4. 発表年 2022年

1. 発表者名 Shiyu Yang, Tetsuya Kanda, Davide Pizzolotto, Daniel M. German, Yoshiki Higo
2. 発表標題 An Empirical Study of Python Code Snippets with Version Compatibility Issues on Stack Overflow
3. 学会等名 第4回 次世代ソフトウェアエコシステムワークショップ
4. 発表年 2022年

1. 発表者名 栗原 拓己, 嶋利 一真, 神田 哲也, 井上 克郎
2. 発表標題 GitHubプロジェクトに利用されているStack Overflowのコード片の進化パターンの調査
3. 学会等名 情報処理学会第208回ソフトウェア工学研究発表会
4. 発表年 2021年

1. 発表者名 杉森 遼, 伊藤 薫, 神田 哲也, 井上 克郎
2. 発表標題 再利用されたライブラリに対するバージョン検出を利用した脆弱性検知ツール
3. 学会等名 ソフトウェアエンジニアリングシンポジウム2021
4. 発表年 2021年

1. 発表者名 嶋利 一真, 石尾 隆, 神田 哲也, 井上 克郎
2. 発表標題 準網羅的な実行トレースを用いた実行差分の可視化とデバッグの有用性検証
3. 学会等名 第2回次世代ソフトウェアエコシステムワークショップ
4. 発表年 2021年

1. 発表者名 栗原 拓己, 嶋利 一真, 神田 哲也, 井上 克郎
2. 発表標題 OSSに利用されているStack Overflowのコード片の進化パターンの調査
3. 学会等名 第2回次世代ソフトウェアエコシステムワークショップ
4. 発表年 2021年

1. 発表者名 嶋利 一真, 石尾 隆, 神田 哲也, 井上 克郎
2. 発表標題 限られた保存領域を使用する準網羅的な実行トレース記録手法の評価
3. 学会等名 第3回次世代ソフトウェアエコシステムワークショップ
4. 発表年 2022年

1. 発表者名 Liang Qiu, Kazumasa Shimari, Tetsuya Kanda, Katsuro Inoue
2. 発表標題 Investigation on the Impact of Logging Configuration Change on the Log Output
3. 学会等名 第3回次世代ソフトウェアエコシステムワークショップ
4. 発表年 2022年

1. 発表者名 栗原 拓己, 嶋利 一真, 神田 哲也, 井上 克郎
2. 発表標題 再利用されたStack Overflowのコード片の進化パターンとOSSでの追従状況の調査
3. 学会等名 第3回次世代ソフトウェアエコシステムワークショップ
4. 発表年 2022年

1. 発表者名 Mohan Bian, Tetsuya Kanda, Kazumasa Shimari, Katsuro Inoue
2. 発表標題 Investigating the impact of source code metrics on merge conflict resolution judgement model
3. 学会等名 情報処理学会第210回ソフトウェア工学研究発表会
4. 発表年 2022年

1. 発表者名 石田 直人, 神田 哲也, 嶋利 一真, 井上 克郎
2. 発表標題 言語サーバを応用した細粒度編集履歴収集プラットフォームの構想
3. 学会等名 ソフトウェアエンジニアリングシンポジウム2020 ワークショップ5
4. 発表年 2020年

1. 発表者名 Koki Ogasawara, Tetsuya Kanda, Katsuro Inoue
2. 発表標題 On the Variations and Evolutions of API Usage Patterns: Case Study on Android Applications
3. 学会等名 The 3rd International Workshop on Software Health (SoHeal 2020) (国際学会)
4. 発表年 2020年

1. 発表者名 白木 秀弥, 神田 哲也, 井上 克郎
2. 発表標題 機械学習による開発履歴のメタ情報を用いたマージコンフリクトの解消パターン判定モデル
3. 学会等名 電子情報通信学会ソフトウェアサイエンス研究会2020年3月研究会
4. 発表年 2020年

1. 発表者名 原口公輔, 神田哲也, 井上克郎
2. 発表標題 ソースコードメトリクスを用いたプログラミングコンテストの類似解答群の検出
3. 学会等名 電子情報通信学会ソフトウェアサイエンス研究会2019年10月研究会
4. 発表年 2019年

1. 発表者名 Kazumasa Shimari, Takashi Ishio, Tetsuya Kanda, Katsuro Inoue
2. 発表標題 Near-Omniscient Debugging for Java Using Size-Limited Execution Trace
3. 学会等名 35th IEEE International Conference on Software Maintenance and Evolution (国際学会)
4. 発表年 2019年

1. 発表者名 Jonathan Komala, Tetsuya Kanda, Tsuyoshi Mizouchi, Katsuro Inoue
2. 発表標題 Visualizing Phase Transition for Real-Time 3D Profiler
3. 学会等名 10th International Workshop on Empirical Software Engineering in Practice (国際学会)
4. 発表年 2019年

〔図書〕 計0件

〔産業財産権〕

〔その他〕

didiff https://github.com/tetsuyakanda/didiff PyVerDetector https://github.com/ysy-dlg/PyVerDetector

6. 研究組織		
氏名 (ローマ字氏名) (研究者番号)	所属研究機関・部局・職 (機関番号)	備考

7. 科研費を使用して開催した国際研究集会

〔国際研究集会〕 計0件

8 . 本研究に関連して実施した国際共同研究の実施状況

共同研究相手国	相手方研究機関
---------	---------