

後方互換性の損失に伴うテストコード変更内容の分類

前川 大樹[†] 伊原 彰紀[‡] 大森 楓己[§] 才木 一也[¶]
和歌山大学[†] 和歌山大学[‡] 和歌山大学[§] 和歌山大学[¶]

1 はじめに

ソフトウェア開発では、開発効率向上のためにライブラリと呼ばれるプログラムを再利用することがある。ライブラリは新機能やバグ修正のために頻繁に更新されるため、ライブラリ利用者は適宜アップデートをする必要がある。しかし、ライブラリの更新には後方互換性を損失する変更（破壊的変更）が含まれることがあり、利用者のプログラムの実行エラーにつながる可能性がある。そのため、ライブラリ開発者はライブラリを更新する際、破壊的変更有無を判断し、適切なバージョン名を付与することで、破壊的変更有無をライブラリ利用者に伝達する。しかし、破壊的変更有無の確認は手動であるため、ライブラリ開発者が破壊的変更を見逃してしまい、意図せず利用者のプログラムが破壊されてしまうことがある。従来研究 [?] では、ライブラリに破壊的変更を加えたりリリースバージョンではテストコードも修正すると考え、テストコードの変更有無による破壊的変更の検出手法を提案した。しかし、テストコードの変更はライブラリ変更に関係するとは限らないため、破壊的変更を誤検出することがある。本研究では、破壊的変更に伴うテストコード変

更内容の分類に取り組み、テスト変更内容に基づく破壊的変更の検出精度を向上する。

2 分析

2.1 データセット

本研究では、松田らが npm から収集した JavaScript ライブラリのリリースバージョンの組 2,111 件 [?] の中から、npm に公開され、ライブラリテストに変更があり、テスト変更の diff サイズが 1024*2 バイト以下であるもの 1,027 件进行分析する。

2.2 分析手法

対象データ 1,027 件から実際に破壊的変更が含まれるバージョンの組と含まれない組 50 件ずつをランダムに抽出して目視し、どのようなテスト変更があったかを調査した。その上で、テスト変更 x を含むテスト変更と x を含まないテスト変更では実際に破壊的変更が含まれる確率が異なると考え、2 群の比率の差の検定を行った。 **TODO:** [検定の内容はやってみてからもっと具体的に書く]

2.3 結果

表 1 は、テスト変更の分類別に、実際の破壊的変更有無、統計的優位差の有無を示す。 **TODO:** [なにか違いが見られたら、その内容とその原因として考えられるものを述べる]

3 おわりに

本研究は、テスト変更の内容分類に取り組み、テスト変更内容と破壊的変更有無の関係性を統

A classification of test code changes accompanied by a lack of backward compatibility

[†] Daiki Maekawa, Wakayama University

[‡] Akinori Ihara, Wakayama University

[§] Fuki Omori, Wakayama University

[¶] Kazuya Saiki, Wakayama University

表 1: **TODO:** [こういうイメージの表を作る]

	破壊的変更あり	破壊的変更なし	有意差
test 変更 A	x x 件	x x 件	無
test 変更 B	x x 件	x x 件	無
test 変更 C	x x 件	x x 件	有
test 変更 D	x x 件	x x 件	無

計的に分析した．分析の結果，**TODO:** [XX
 がわかった]．今後は，テスト変更内容を自動
 で分類することで標本数を増やすとともに，破
 壊的変更の内容についても調査し，どのよう
 な破壊的変更の時にテストの分類が有効で，ど
 のような破壊的変更で有効ではないかを調査
 する．

謝辞

TODO: [謝辞]

TODO: [overleaf のせいで参考文献のフ
 ォーマットがあれなので直す]