

Fault-Prone モジュール予測における 設計メトリクス値の変化履歴の検討

遠藤 直人[†] 野中 誠[†]

東洋大学経営学部[†]

1. はじめに

Fault-Prone (以降 FP) モジュール予測は、これまでに多くの研究がなされている。近年では、オープンソースソフトウェアのリポジトリ情報を活用した研究が盛んである。しかし、それらのリポジトリには、ソースコードファイルがどのような原因で修正されたのかの情報が十分に蓄積されていないことが多い。そのため当該ソフトウェア自体に欠陥があったために修正されたのか、外部の連携システムに欠陥があったために当該ソフトウェアを修正せざるをえなかったのかが考慮されておらず、ファイルの修正を一元的に扱って分析している研究事例も多い。このような分析により得られた FP モジュール予測モデルをもとに開発者に重点レビュー実施などの対応を求める場合、モデルの納得性が低いために、開発者への訴求度が低くなってしまふ。

また、FP モジュール予測に関する先行研究では、設計メトリクスの値が定まった時点と、欠陥が混入した時点の前後関係を緻密に解析せずに、最新バージョンのソフトウェアから設計メトリクスの値を測定しているものもある。この場合、設計メトリクスを説明変数としていながら、その測定時点が、目的変数に関わる事象である欠陥混入よりも後になってしまう。ただし、異なるバージョン間で設計メトリクスの値に大きな差異がなければ、このような前後関係を緻密に分析せずに最新バージョンの設計メトリクスを測定することで、FP モジュール予測において有用な設計メトリクスを特定できる。これは、FP モジュール予測分析における費用対効果の面でメリットとなる。

本稿では、設計メトリクスの値が定まった時点と欠陥が混入した時点の前後関係を、設計メトリクスの変化履歴から解析した上で、先行研究[1]の予測モデルの評価を行う。

2. 分析対象と設計メトリクス値の変化履歴

分析対象のデータとして、オープンソースソフトウェアである Twitter4J (記述言語 Java, 総行数 20,385 行, 総ファイル数 162 個) の、2007 年から 2009 年までの 29 バージョンのソースコードと、各バージョンのリリース後の修正情報を用いた。設計メトリクス値の測定には、測定ツール Eclipse Metrics plug-in 1.3.6[2] と Chidamber & Kemerer Java Metrics Tool[3] を使用した。

設計メトリクスの値が定まった時点と欠陥が混入した時点の前後関係を調べるため、欠陥修正がなされたファイル (総ファイル数 79 個) について、そのファイルが修正された時点でのメトリクス値、修正された時点のひとつ前のバージョンのメトリクス値、最新バージョン時点でのメトリクス値の合計 3 つの値をそれぞれのファイルごとに測定した。なお、ひとつのファイルに複数の欠陥修正行っている場合、計測タイミングが増える。

その結果、修正された時点とそのひとつ前の時点でメトリクスの値が変化していたファイルもごく少数あったが、ほとんどのファイルは最初から最後まで設計メトリクスの値が一定であった。そのため、今回の分析対象においては、欠陥が混入した時点を緻密に分析せずに、最新バージョンにおける設計メトリクスの値を測定して FP モジュール予測を行っても、分析結果に影響を与える可能性は低いことがわかった。

3. 先行研究の妥当性評価

野中らの研究[1]では、本研究と同じ分析対象のデータを用いて、欠陥分類別の FP モジュール予測モデルを構築している。この研究では、最新バージョンの設計メトリクスを測定して分析していたが、2 章で示した結果から、その分析の妥当性は確保されていると考える。

表 1 に、野中らの研究[1]で示された予測モデルのうち、欠陥分類がなされていないものを示す。表 1 では、LCOM-HS (凝集度の欠如度合い, 文献[4]による定義), WMC (複雑度の重み付きメ

A Study on Fault-Prone Module Prediction in Consideration of Change History of Design Metrics

[†]Naoto Endo and Makoto Nonaka, Faculty of Business Administration, Toyo University

表1 FP モジュール予測モデル

説明変数	分類なし
定数項	-2.11(0.000)
LCOM-HS[4]	2.14(0.015)
WMC	0.0746(0.017)
CBO	0.212(0.008)

ソッド数), CBO (オブジェクト間結合度) の3つのメトリクス値から FP モジュール予測モデルを構築している。

しかし, このモデルの評価についてまだ言及すべき部分が2点ある。1つは, モデル式から予測した値が, 欠陥の修正有無の実績においてどのように分布していたかが示されていないという問題である。もう1つは, モデル自体の妥当性が十分に評価されていないことである。以上のことを先行研究[1]の課題として, 本研究で取り組む。なお, FP モジュール予測の説明変数として, オブジェクト指向設計メトリクスとして使われる C&K メトリクス[4]を使用している。

4. 予測モデルの評価

実績の修正の有無を横軸に, モデル式による予測値を縦軸にとり, これらの関係を箱ひげ図により示したものが図1である。これを数値化し, クロス集計したものが表2である。表2から, 修正ありのところだけをみると, モデルから予測した60のファイルのうち48のファイルにおいて実際に修正が行われている。よって80%の再現率と適合率があるといえる。

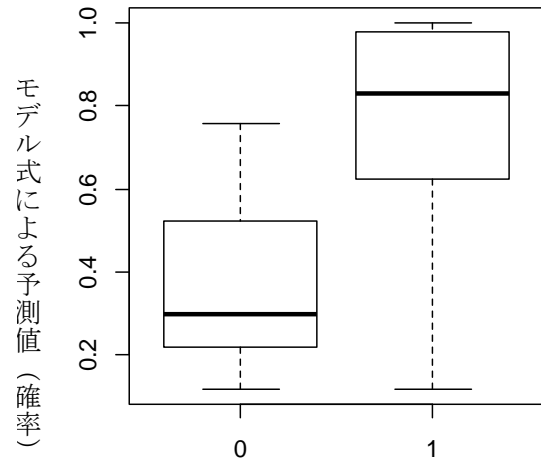
また, モデル自体の評価をするために, 交差検証 (Leave One Out) を行った。その結果, 予測誤差16.3%という結果が得られた。これらの結果から, 野中らの研究[1]が導き出した予測モデルの式は, 最新バージョンに対して設計メトリクスを測定する簡便な方法であったが, 設計メトリクスの変化履歴を緻密に考慮しなくても, 高い予測精度のモデルであったことが示せた。

5. おわりに

本稿の報告では, FP モジュール予測を行う際に, 欠陥発生時に使われていた状態のデータを用いるのではなく, 最新のデータを用いても大きな影響はないといえる。よって比較的手軽に Fault-Prone 予測できることを示すことができた。しかし, 一事例の評価であるためにサンプルサイズも小さく, 設計メトリクス値が大きく変化しないという知見がほかの事例でも当てはまるかどうかは不明である。この点について, さらなる検証を行う必要性がある。

表2 修正有無の予測精度

	修正なし(予測)	修正あり(予測)
修正なし(実績)	30	12
修正あり(実績)	12	48



実績の修正有無 (0 = なし, 1 = あり)

図1 修正有無の予測分布図

参考文献

- [1] 野中誠, 仙頭洋一: 欠陥分類を用いた Fault-Prone モジュール予測の適用事例, ウインターワークショップ2012・イン・琵琶湖論文集, 情報処理学会 (to appear) .
- [2] <http://sourceforge.net/projects/metrics>
- [3] <http://www.spinellis.gr/sw/ckjm>
- [4] Henderson-Sellers, B.: *Object-Oriented Metrics: Measures of Complexity*, Prentice-Hall (1996).
- [5] Chidamber, S. R. and Kemerer, C. F.: A Metrics Suite for Object-Oriented Design, *IEEE Trans. on Software Eng.*, Vol. 20, No. 6, pp.476-493 (1994).