

Comparing Quality Metrics for Cloned and non cloned Java Methods: A Large Scale Empirical Study

Author:
Vaibhav Saini
Hitesh Sajnani
Cristina Lopes

2016 IEEE International
Conference on Software
Maintenance and Evolution
(ICSME), 256-266, 2016

目次

- 背景, 目的
- アプローチ
- 実験結果
- まとめ, 所感

背景, 目的

背景

- これまでクローンが有害であるかどうかの研究が多くなされてきた
- クローンのこういった点が有害かそうでないかを更に知りたい
- クローンに対してメトリクスを計測してクローンの有害性を分析した研究はまだない

目的

- クローンのメソッドとそうでないメソッドのメトリクスの違いがあるかどうかを分析する
- クローンのメソッドはそうでないメソッドと比べてどのようなメトリクスが有害な値を示すかを調べる

アプローチ

アプローチ

- 4,421個のJavaのオープンソースプロジェクトのクローンのメソッドとそうでないメソッドに対して以下のメトリクスを計測する
 - 複雑さに関するメトリクス
 - モジュール性に関するメトリクス
 - コメントに関するメトリクス
- クローンのメソッドとそうでないメソッドの間でそれぞれのメトリクスに有意差があるかどうかをウィルコクソン順位和検定を用いて調べる

リサーチクエストション

- RQ1: クローンのメソッドはそうでないメソッドと比べて複雑さに関するメトリクスの値が小さいか
- RQ2: クローンのメソッドはそうでないメソッドと比べてモジュール性に関するメトリクスの値が小さいか
- RQ3: クローンのメソッドはそうでないメソッドと比べてコメントに関するメトリクスの値が小さいか

使用するメトリクス一覧

矢印が下を指しているときはそのメトリクスの値が小さいほうが有益であることを示している
上を指しているときは同様にメトリクスの値が大きいほうが有益であることを示している

Category		Name	Description
Code Complexity	↓	COMP	McCabes cyclomatic complexity
	↓	NOA	Number of arguments
	↓	VDEC	Number of variables declared
	↓	VREF	Number of variables referenced
	↓	NOS	Number of statements
	↓	NEXP	Number of expressions
	↓	MDN	Method, Maximum depth of nesting
	↓	HLTH	Halstead length of method
	↓	HVOC	Halstead vocabulary of method
	↓	HVOL	Halstead volume
	↓	HDIF	Halstead difficulty to implement a method
	↓	HEFF	Halstead effort to implement a method
	↓	TDN	Total depth of nesting
	↓	CAST	Number of class casts
	↓	LOOP	Number of loops (for,while)
	↓	NOPR	Total number of operators
Modularity	↓	NAND	Total number of operands
	↓	HBUG	Halstead prediction of number of bugs
	↓	NLOC	Number of lines of code
	↓	CREF	Number of classes referenced
	↓	XMET	External methods called by the method
Documentation	↑	LMET	Local methods called by the method
	↑	EXCR	Number of exceptions referenced by the method
	↑	EXCT	Number of exceptions thrown by the method
	↑	MOD	Number of modifiers
Documentation	↑	NOC	Number of comments
	↑	NOCL	Number of comment lines

Halsteadメトリクス

n_1 :オペレータの種類数

n_2 :オペランドの種類数

N_1 :オペレータの個数の合計

N_2 :オペランドの個数の合計

$HLTH: N_1 + N_2$

$HVOC: n_1 + n_2$

$HVOL: HLTH \log_2 HVOC$

$HDIF: \frac{n_1}{2} \times \frac{N_2}{n_2}$

$HEFF: HDIF \times HVOL$

$HBUG: \frac{HEFF^2}{3000}$

TABLE I: Software Quality Metrics

実験結果

サイズ調整前の結果

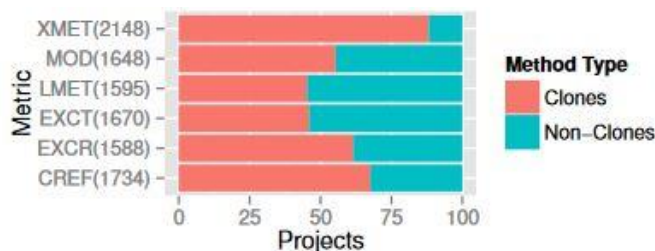


Fig. 5: Wilcoxon Rank-Sum test: stacked bar chart comparing cloned and non cloned methods. The Vertical-axis shows the *Modularity* metrics along with the number (in parenthesis) of subject projects that showed significant difference between cloned and non cloned methods ($p < 0.05$, $r > 0.1$).

RQ2の結果

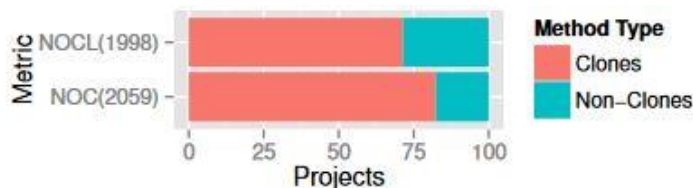


Fig. 6: Wilcoxon Rank-Sum test: stacked bar chart comparing cloned and non cloned methods. The Vertical-axis shows the *Documentation* metrics along with the number (in parenthesis) of subject projects that showed significant difference between cloned and non cloned methods ($p < 0.05$, $r > 0.1$).

RQ2の結果

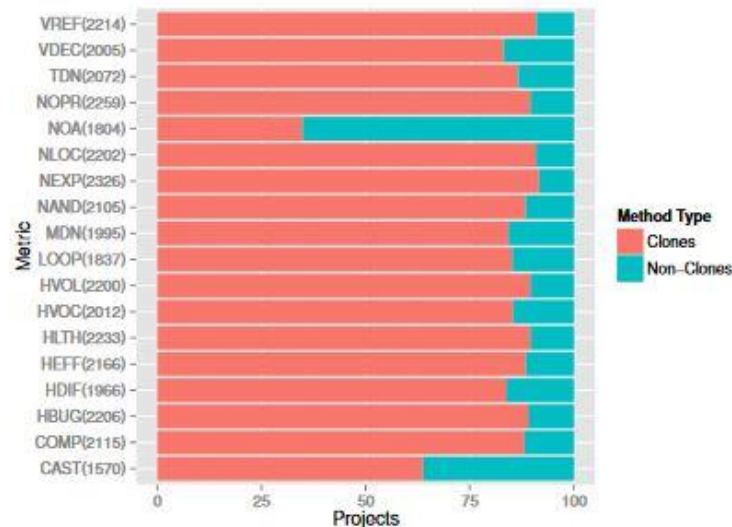


Fig. 4: Wilcoxon Rank-Sum test: stacked bar chart comparing cloned and non cloned methods. The Vertical-axis shows the *Complexity* metrics along with the number (in parenthesis) of subject projects that showed significant difference between cloned and non cloned methods ($p < 0.05$, $r > 0.1$).

RQ1の結果

赤いバーはクローンメソッドのメトリクスがよりよい値を示したプロジェクトの個数を示している
青いバーはクローンでないメソッドのメトリクスがよりよい値を示したプロジェクトの個数を示している

サイズの調整

- それぞれのプロジェクトでクローンメソッドとそうでないメソッドの個数を合わせる
- クローンメソッドのNOSの中央値はそうでないメソッドと比べて20%小さい
- NOSによる影響を除去するためにすべてのメトリクスをNOS(文の数)で割る
 - COMPが10でNOSが50の場合, $10/50=0.2$
- さらにNOSの影響を除去するために線形回帰モデルを用いる

NOSで割った場合

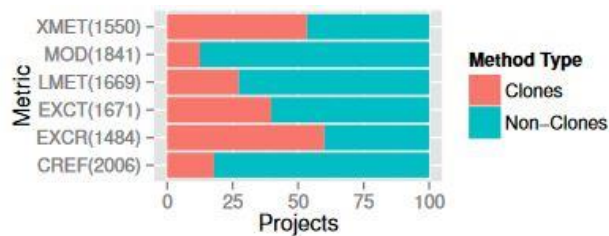


Fig. 8: Wilcoxon Rank-Sum test: stacked bar chart comparing cloned and non cloned methods. The Vertical-axis shows the normalized *Modularity* metrics along with the number (in parenthesis) of subject projects that showed significant difference between cloned and non cloned methods ($p < 0.05$, $r > 0.1$).

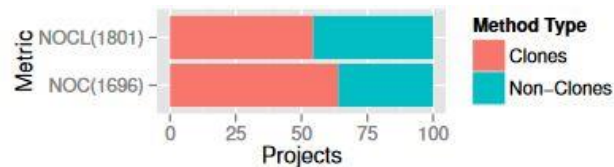


Fig. 9: Wilcoxon Rank-Sum test: stacked bar chart comparing cloned and non cloned methods. The Vertical-axis shows the normalized *Documentation* metrics along with the number (in parenthesis) of subject projects that showed significant difference between cloned and non cloned methods ($p < 0.05$, $r > 0.1$).

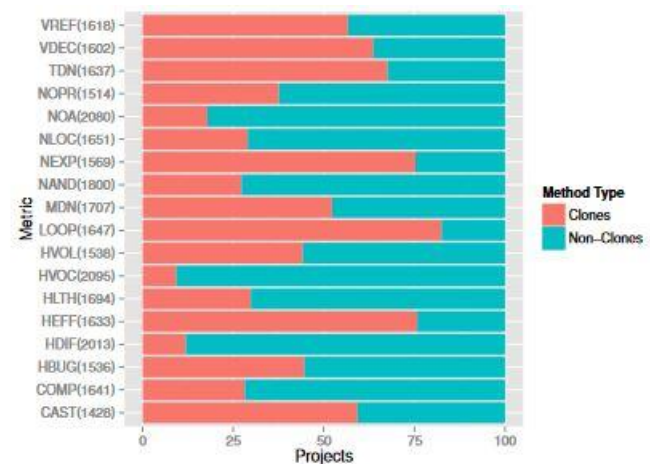


Fig. 7: Wilcoxon Rank-Sum test: stacked bar chart comparing cloned and non cloned methods. The Vertical-axis shows the normalized *Complexity* metrics along with the number (in parenthesis) of subject projects that showed significant difference between cloned and non cloned methods ($p < 0.05$, $r > 0.1$).

線形回帰モデルの利用

- 多くのメトリクスがNOSと相関関係があり, NOSで割ってもNOSの影響を除去し切れない
- 線形回帰モデルを用いてクローンメソッドのメトリクスとそうでないメソッドのメトリクスの2つの回帰直線の式を出し, どちらの直線の切片が大きいかでクローンとそうでないメソッドのメトリクスの無害さの優劣を決める

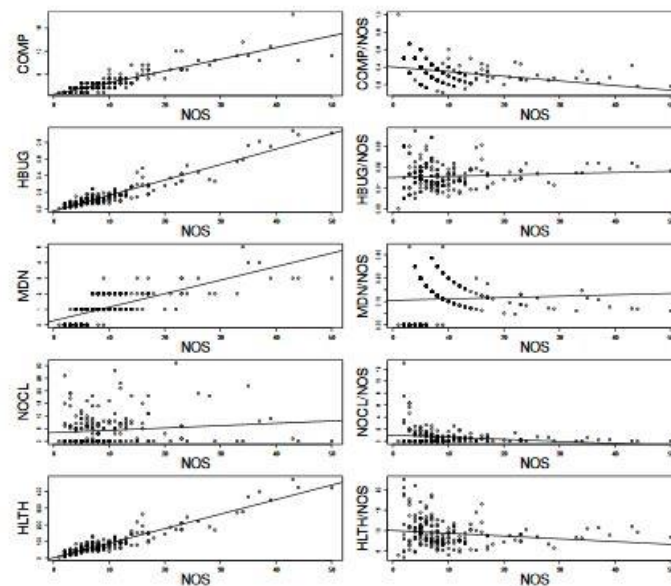
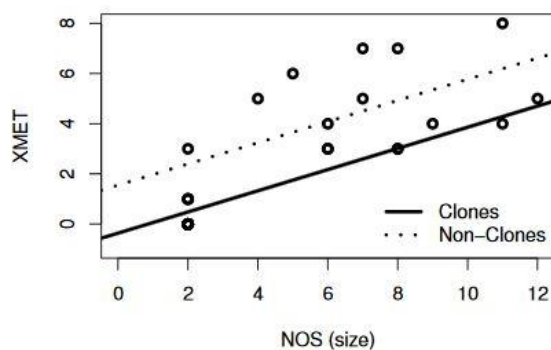


Fig. 10: Scatter plots showing relationship between quality metrics on vertical-axis and size metric (NOS) on horizontal-axis, before and after carrying out normalization of metrics. The plots in the left column show the relationship between the metric and the size before normalization, whereas the plots on the right show the relationship between the normalized metrics and the size metric.

線形回帰モデルを用いた例

- XMETを例として用いる
- $E(XMET) = b_0 + b_1 \times NOS + b_2 \times isClone$
 - isClone: クローンメソッドかどうかを示す値. 0か1になる
 - b_0, b_1, b_2 : 線形回帰モデルのパラメータ
- $E(XMET) = 1.5510 + 0.42 \times NOS - 1.91 \times isClone$
- クローンメソッドでない場合: $E(XMET) = 1.55 + 0.42 \times NOS$
- クローンメソッドの場合: $E(XMET) = -0.360251 + 0.42 \times NOS$



クローンでないメソッド
のXMETの方が大きい
ことがわかる

Fig. 11: Fitted regression lines for cloned and non cloned methods showing average difference between their XMET values. The vertical-axis shows the XMET metric values and the horizontal-axis shows the size of the methods as measured using NOS metric.

線形回帰モデルを用いた場合

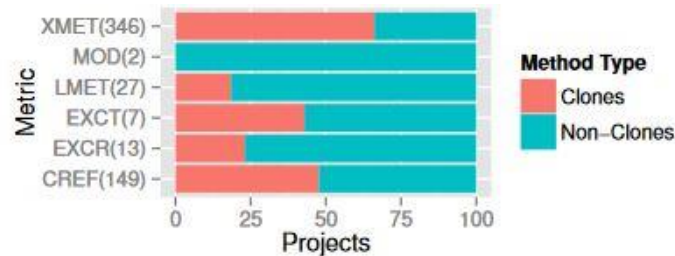


Fig. 13: Regression: stacked bar chart comparing cloned and non cloned methods. The Vertical-axis shows the *Modularity* metrics along with the number (in parenthesis) of subject projects that showed significant difference between cloned and non cloned methods ($p < 0.05$, $r > 0.1$).

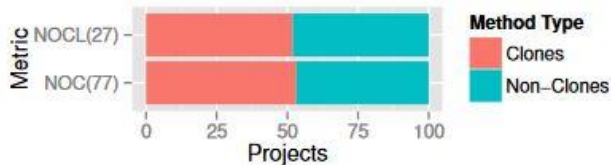


Fig. 14: Regression: stacked bar chart comparing cloned and non cloned methods. The Vertical-axis shows the *Documentation* metrics along with the number (in parenthesis) of subject projects that showed significant difference between cloned and non cloned methods ($p < 0.05$, $r > 0.1$).

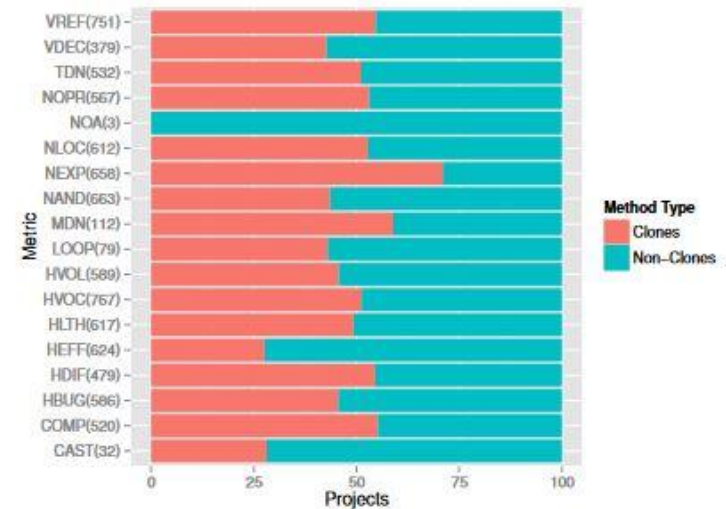


Fig. 12: Regression: stacked bar chart comparing cloned and non cloned methods. The Vertical-axis shows the *Complexity* metrics along with the number (in parenthesis) of subject projects that showed significant difference between cloned and non cloned methods ($p < 0.05$, $r > 0.1$).

結果の要約

- 17%のプロジェクトでメトリクスに有意差があり, いくつかのメトリクスが少しだけ影響を与えていることがわかった
- 無害なメトリクスの値を持つクローンメソッドを多く含むプロジェクトの数と無害なメトリクスの値を持つクローンでないメソッドを多く含むプロジェクトの数は大体同じ
- クローンメソッドのNOSの中央値はそうでないメソッドと比べて20%小さい
- NEXPとXMETが30%以上のプロジェクトでクローンメソッドの方が低い値(無害な値)になることがわかった

まとめ, 所感

まとめ

- クローンメソッドの方がそうでないメソッドと比べてNOSが小さいことがわかり、開発者はなるべく小さいメソッドを複製することを心がけていることがわかった
- それ以外のほとんどのメトリクスはクローンメソッドとそうでないメソッドとであまり差がないことが分かった
- NEXPとXMETはクローンメソッドのほうが少なくとも30%のプロジェクトで小さい値であることがわかった

所感

- 4,000以上のプロジェクトを扱ったり、メソッドのサイズの影響を除去したりなど、かなり労力がかかった研究だと思った
- ほとんどのメトリクスでクローンメソッドとそうでないメソッドの間に差がないということが分かり、改めてクローンの無害さを実感した
- クローンの中でもタイプ別やバグ票と結びついたものなどに絞ればメトリクスの値が有害性を示すかも知れないと思った