

What Makes a Good Commit Message?

Yingchen Tian, Yuxia Zhang, Klaas-Jan Stol, Lin Jiang, Hui Liu

44th International Conference on
Software Engineering (ICSE2022)

発表者：松尾 和輝

背景

- 良いコミットメッセージは、変更の文脈を共同作業者に伝え、共同作業者が変更を確認し、その影響を理解するために必要である[2, 69]
 - コミットメッセージは、プロジェクトに加えられた変更を理解するための情報源になりうる
- 時間の不足やモチベーション不足のため、コミットメッセージの品質が低下する
 - 良いコミットメッセージを書きたくても、何が良いかが開発者にはわからない

背景と目的

- コミットメッセージを自動生成するツールはいくつか提案されているが、良くないコミットメッセージを用いて訓練、評価してしまっている
 - 何を変更したかを生成できても、なぜ変更したかを生成できない
- 目的
 - 良いコミットメッセージに関する調査を行う
 - 良いコミットメッセージ：WhyとWhatの両方が書かれたメッセージ

リサーチクエスチョン

- RQ1
 - 良くないコミットメッセージはどの程度存在するのか？
- RQ2
 - 良いコミットメッセージのパターンとは？
- RQ3
 - 良いコミットメッセージを識別できるか？

対象プロジェクト

- 活発で人気のあるプロジェクト
- 産業界で広く使用されているJavaで書かれたプロジェクト
- 過去にコミットメッセージの自動生成に焦点を当てた研究の対象となったプロジェクト

を満たす5つのプロジェクトを対象プロジェクトとする

Table 1: Dataset summary statistics (until Feb 2021)

Project	Start	#Contrib.	#Commits	#Cleaned
Spring-boot	Oct-2012	812	30,072	21,169
Apache Dubbo	Jun-2012	404	2,687	2,249
Okhttp	Jul-2012	236	4,800	2,817
Junit4	Apr-2009	151	2,467	2,035
Retrofit	Sep-2010	152	1,860	1,078
			41,886	29,348

データ収集

- 5つのプロジェクトから
 - 英語で書かれている
 - 人が書いている (botによって書かれていない)

コミットメッセージ29,348件を収集

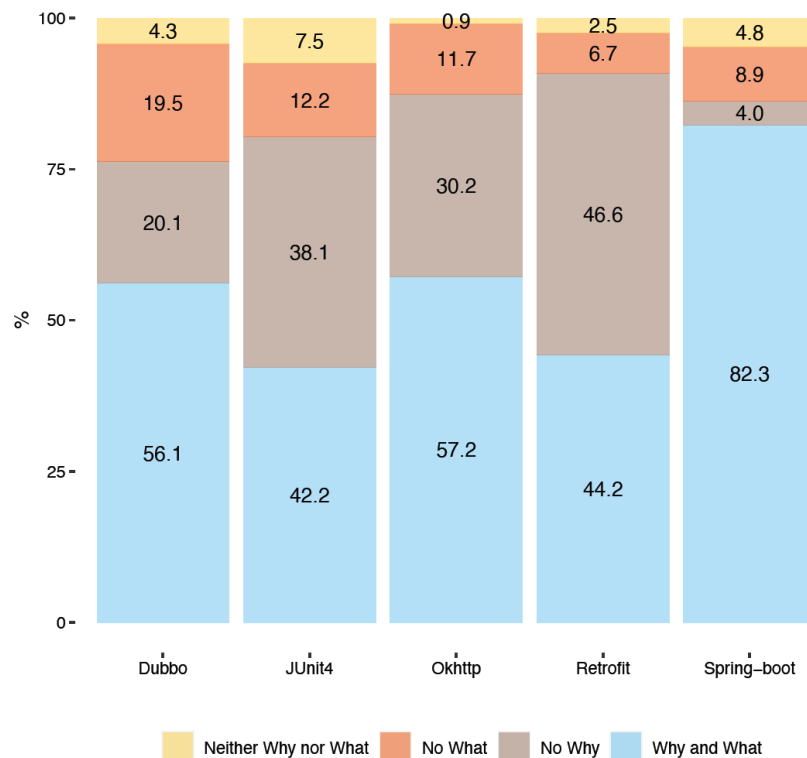
- クラスター無作為抽出法[5]を用い, 339人の開発者がコミットした1,649件のコミットメッセージを収集
- 複数の変更 (例: **Clean Up**と機能追加) が行われた52件のコミットメッセージを除いた1,597件をデータセットとする

RQ1：アプローチ

- RQ1
 - 良くないコミットメッセージはどの程度存在するのか？
- 1,597件のコミットメッセージをラベリングし，以下の4つのタイプに分類する
 - 良いコミットメッセージ
 - Why and What
 - 良くないコミットメッセージ
 - No Why
 - No What
 - Neither Why nor What

RQ1：結果

- 良くないコミットメッセージは、平均で**44%**であった
 - 既存の自動コミットメッセージ生成ツールでは良くないコミットメッセージを半分も含むデータセットで訓練，評価してしまっている



RQ1：結果

- 「Neither Why nor What」のコミットメッセージ65件を調査
- 以下の5つの特徴が見られた
 - 1単語で記述されたメッセージ
 - コミットであること以外を表現しないメッセージ
 - 変更の範囲以外を説明しないメッセージ
 - コードの差分から容易に推測できる内容を記述したメッセージ
 - 変更に関係なことを記述したメッセージ
- これらのコミットメッセージを特定，除去することは容易であるが，「No Why」，「No What」のコミットメッセージを特定するのは容易でない

RQ2：アプローチ

- RQ2
 - 良いコミットメッセージのパターンとは？
- 「Why and What」のコミットメッセージ**271**件を抽出
- このうち、プルリクエストや課題報告へのリンクのみで**Why**を表現している**19**件を削除
- 残った**252**件に対して、**Why**と**What**の情報をどのように表現しているか調査しカテゴリ分類する

RQ2：アプローチ

- 開発者が**Why**と**What**をコミットメッセージで表現する方法は、保守活動の種類によって異なる可能性がある
- 保守活動の種類も分類し、コミットメッセージの表現カテゴリと保守活動の関係を調査
- 保守活動の種類として**Mockus**と**Votta**の定義[49]を採用
 - **Corrective** :
 - 処理、性能、実装の不具合に対応するもの
 - **Adaptive** :
 - データ環境や処理環境の変更を表すもの
 - **Perfective** :
 - 効率、パフォーマンス、クリーンアップなどの非機能属性の改善を行うもの

RQ2：結果

- Why表現のカテゴリ
 - 以下の5つに分類
- Describe issue (27.8%)
 - 問題を説明するもの
- Illustrate requirement (18.3%)
 - このコミットに至った要求の出所を記述するもの
- Describe objective (8.7%)
 - 目的を記述するもの
- Imply necessity (28.6%)
 - 必要性を暗示するもの
- Missing Why (19.4%)
 - 常識的であったり容易に推測できる場合など理由が必要ないもの

RQ2：結果

- What表現のカテゴリ
 - 以下の4つに分類
- Summarize code object Change（70.6%）
 - コードの変更を要約するもの
- Illustrate function（2.4%）
 - 変更の基盤となる技術的な原則を記述したもの
- Describe implementation principle（25.8%）
 - 機能を説明するもの
- Missing What（7.5%）
 - 何が変更されたのかを簡単に推測できるもの

RQ2：結果

- コミットメッセージの表現カテゴリと保守活動の関係を調査

- 保守活動の種類によって、**Why**の表現方法は異なる
 - 保守活動の種類によっては適したコミットメッセージの書き方が変わる
- **What**の表現方法は、保守活動の種類によっては変わらない

		Corrective (#116)	Adaptive (#63)	Perfective (#73)
How to express "Why"	Describe issue	45.7%	12.7%	6.9%
	Illustrate requirement	12.1%	22.2%	21.9%
	Describe objective	6.9%	7.9%	11.0%
	Imply necessity	19.0%	39.7%	26.0%
	Missing Why	12.1%	15.9%	34.2%
	Describe issue & Describe objective	0.8%	0 %	0 %
	Describe issue & Imply necessity	2.6%	0 %	0 %
	Illustrate requirement & Imply necessity	0.8%	1.6%	0 %
	Total	100.0%	100.0%	100.0%
How to express "What"	Summarize code object Change	58.6%	60.3%	76.7%
	Illustrate function	22.4%	27.0%	8.2%
	Describe implementation principle	4.3%	1.6%	0 %
	Missing What	6.1%	3.2%	13.7%
	Summarize code object change & Illustrate function	8.6%	7.9%	1.4%
	Total	100.0%	100.0%	100.0%

RQ3：アプローチ

- RQ3
 - 良いコミットメッセージを識別できるか？
- 8種類の分類器に対し，10分割交差検証を行う
 - 1,597件のコミットメッセージをランダムに10個のサブセットに分割
 - 1個のサブセットをテストデータに，9個のサブセットを学習用のデータとする
 - 10個のサブセット全てが1回ずつテストデータになるように，10回学習を行い，精度の平均を取る
- Why, Whatが含まれているかを識別するモデルをそれぞれで構築

RQ3：結果

- Bi-LSTMがWhy, Whatの両方で最も精度が良かった

Techniques	Accuracy C-Why	Accuracy C-What
Bi-LSTM	84.7%	91.0%
LSTM	83.6%	90.1%
Logistic Regression	79.1%	85.5%
MLP	80.3%	85.1%
Random Forest	76.5%	86.1%
Gradient Boosting	72.5%	77.1%
KNN	74.5%	70.7%
Decision Tree	68.5%	76.8%

RQ3：結果

- Why, Whatが含まれているかを識別するBi-LSTMベースのモデルを組み合わせて、WhyとWhatが両方あるかどうかを識別するモデルを構築
- 75.9%と良い精度が得られた

- C-Why：Whyがあるかどうか
 - Positive：ない
 - Negative：ある
- C-What：Whatがあるかどうか
 - Positive：ない
 - Negative：ある
- C-Good：WhyとWhatが両方あるかどうか
 - Positive：どちらもある
 - Negative：どちらかが欠けている

Metrics		C-Why	C-What	C-Good
Positive	Precision	76.5%	78.2%	81.6%
	Recall	70.9%	64.5%	74.0%
	F1	73.1%	68.9%	77.6%
Negative	Precision	88.1%	93.4%	70.0%
	Recall	90.2%	96.2%	78.4%
	F1	89.1%	94.7%	73.9%
Accuracy		84.7%	91.0%	75.9%

まとめ

- 目的
 - 良いコミットメッセージに関する調査
- 結果
 - 良くないコミットメッセージは**44%**あった
 - 保守活動の種類によって、**Why**の表現方法は異なる
 - **What**の表現方法は、保守活動の種類によっては変わらない
 - 良いコミットメッセージを識別する分類器で、**75.9%**の精度が得られた

所感

- RQ3のモデル構築の話が，わかりづらかった
 - どうやって2つのモデルを組み合わせたのかが書いてない
 - WhyやWhatがあるかどうかの話で，なぜPositiveをないにするのかがわからない
- WhatやWhyが書かれたコミットメッセージは良いとされるが，どれだけの分量を書けば良いかはわからない
 - 書かなすぎもダメ，書きすぎもダメだと思う
- WhatやWhyが書かれたコミットメッセージが有効かどうかはわからない
 - 良いコミットメッセージと良くないコミットメッセージが，プロジェクトに与える影響に違いがあるのか