

The Value of Effective Pull Request Description

Shirin Pirouzkhah¹, Pavlína Wurzel Gonçalves²,
Alberto Bacchelli¹
¹University of Zurich
²INESC TEC

MSR 2026

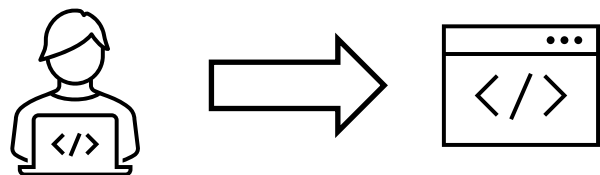
背景：コードレビュープロセス

OSS開発におけるソースコード変更の提案の管理方法

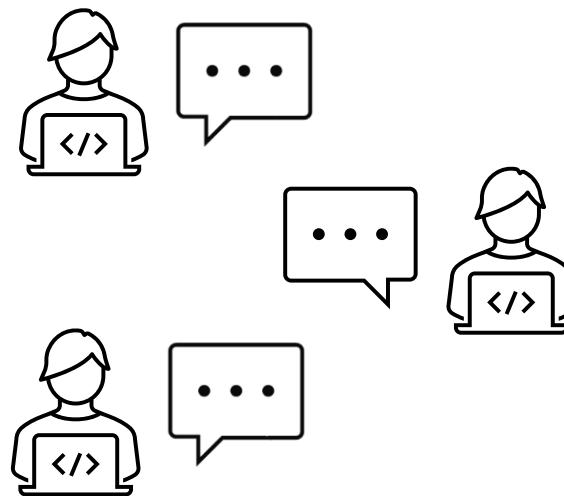
- チケット形式
- GitHubでは、チケットのことをプルリクエスト(PR)と呼ぶ
- PRは投稿からコードレビューを得て、ソフトウェアへ導入される

チケットによるソースコード変更提案の流れ

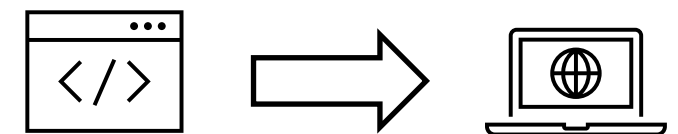
1. チケットの投稿



2. コードレビュー



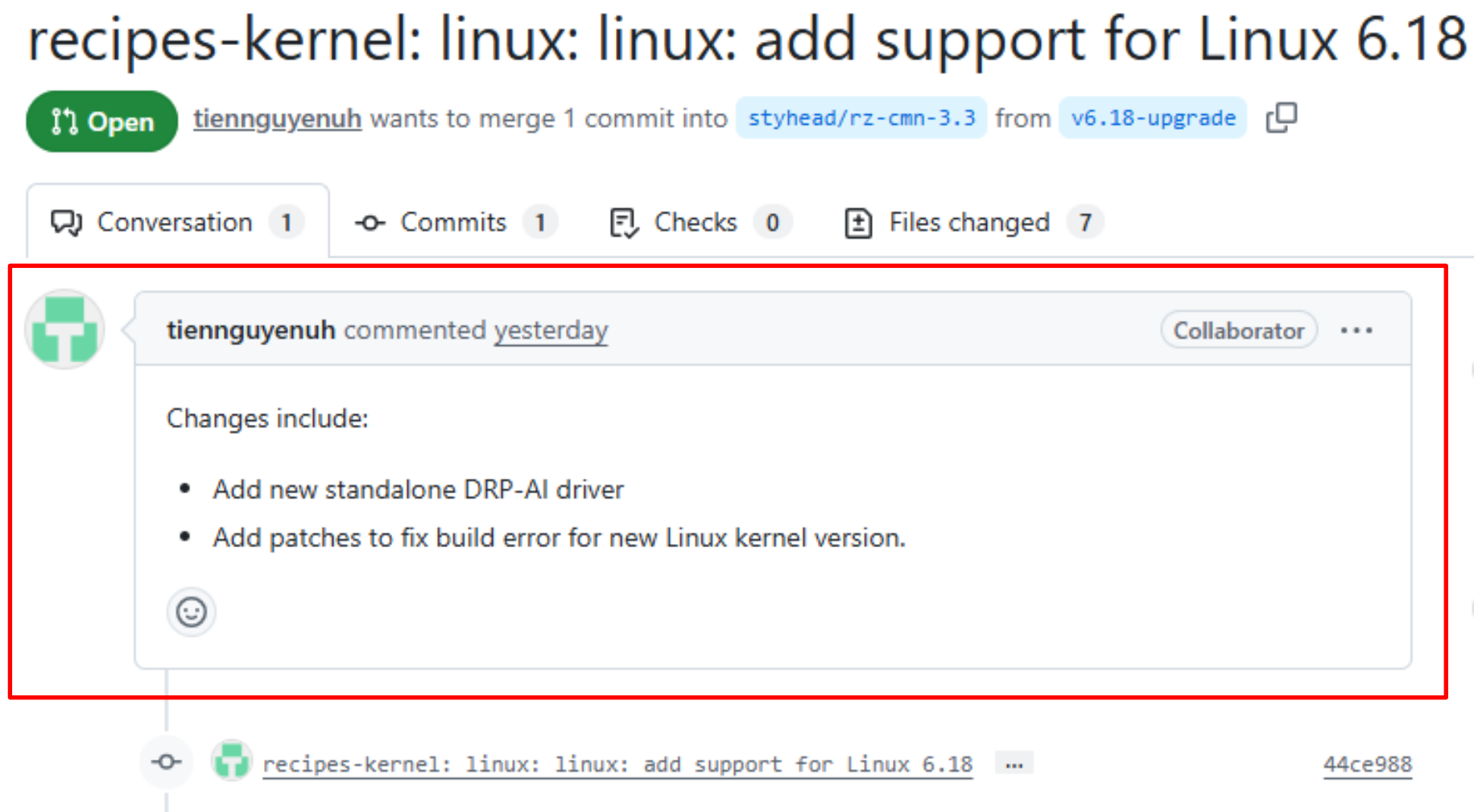
3. ソフトウェアへ導入



背景：PRの説明文

プルリクエストには説明文を付与することができる

- 下図の赤枠



PRの説明文は、レビュアーが提出されたコードの理解するに役立つ

背景：PR説明文がコードレビュープロセスに与える影響

PRの説明文がコードレビュープロセスに及ぼす影響に関する調査

- PR説明文の有無が、マージの可否を初めとしたコードレビュープロセス全般に及ぼす影響の調査[1]
- PR説明文の長さが、マージの可否に与える影響の調査[2]

上記の研究は、いずれも表層的な要素のみ分析している

[1]Sadowski, C., Söderberg, E., Church, L., Sipko, M., & Bacchelli, A. (2018). Modern code review: A case study at Google.

[2]Zhang, X., Yu, Y., Gousios, G., & Rastogi, A. (2022). Pull request decisions explained: An empirical overview.

背景：PR説明文の書き方

様々な文献で、PR説明文のベストプラクティス(いいPRの書き方)が紹介されている

- 例：GitHubの公式HPで「いいPRの説明文の書き方」として、「変更の目的を記載すること」等を紹介している[3]

課題1：各文献がバラバラな表現で異なる要素を推奨しており、統一された枠組みがない

課題2：このような「PRの説明文の書き方」がコードレビュープロセスに与える影響が明らかになっていない

[3] GitHub Docs. 2025. Best Practices for Pull Requests. <https://docs.github.com/en/pull-requests/collaborating-with-pull-requests/gettingstarted/best-practices-for-pull-requests> Accessed

研究目的

目的：

- 効果的なPRの説明文の構成要素の解明
- その構成要素がコードレビュープロセスの結果にどのような影響を与えるかを解明

リサーチクエスション：

- RQ1:PRの説明文の記述に含めることが推奨される要素は何か？
- RQ2:PRの説明文の記述要素の存在は，コードレビューの結果にどのような影響を与えるか
- RQ3:開発者はPRの説明文とその要素の重要性和価値をどのように認識しているか

RQ1:調査対象

文献の収集元

- Google検索
- GitHubDocs等といったコミュニティサイト
- シード頁からの雪だるま式検索

選定基準

- PR説明の書き方を明示的に解説している
- 公開アクセス可能
- 実務者向け
- 個別要素に分解可能な形式のもの
- 除外：PR説明に直接関係しない一般的なコードレビューのアドバイス

選定結果

- 22件のグレー文献（査読のない資料）（業界ガイドライン・ブログ・技術レポート等）

RQ1:分類体系構築

収集した文献から推奨される実践とガイダンス項目を抽出

オープンコーディングと反復分類により分類体系構築

オープンコーディングとは

- 事前にカテゴリを決めずにデータを繰り返し読みながら意味のある単位にラベルを付け、類似するものをまとめてカテゴリを帰納的に導出する質的分析手法

反復分類とは

- 初回のコーディング結果を見直してカテゴリの統合・分割・再定義を繰り返し、最終的に安定した分類体系に収束させるプロセス

RQ1:結果

要素	説明
Purpose	変更の目的（何をしたか）
Reason	変更の理由（なぜしたか）
Code Explanation	変更コードの具体的な説明
Link	関連 Issue ・ チケットへのリンク
Feedback Type	レビュアーに求めるフィードバックの種類
Order	レビューを読む順序の案内
Test Explanation	テスト内容 ・ 実施済みテストの説明
Screenshots	視覚的証拠（UI変更等のスクリーンショット）

RQ2:調査対象(1/2)

プロジェクトの収集元:GHTorrentデータセット

- 約11,000件のプロジェクトの330万件のClosedPRが収録されている

選定基準(プロジェクト)

- プロジェクトに含まれるPRの三分の一が外部貢献者により提出されている
 - 理由：コアメンバーが支配するプロジェクトはPRの説明文が簡素になりがち→PRの諸要素が検出されづらい
 - 外部貢献者：push権限を持たないユーザー
 - コアメンバー：push権限を持つユーザー
- プロジェクトにテストファイルが含まれている
- プロジェクトは200PR以下

選定結果

- 156プロジェクト

RQ2:調査対象(2/2)

選定基準(プルリクエスト)

- P Rの投稿者以外がレビューしている
 - 理由：自己レビューP Rを排除するため
- 少なくとも1つ以上のソースコードファイルを変更している
 - 理由：ドキュメントの変更のみといったP Rを除外するため

選定結果

- 8万件のGitHub PR

RQ2: データ加工

GitHubAPIを活用して、各PRのPR説明文を取得

取得した説明文に各PR記述要素が含まれているかどうかをLLama 3.1-70Bに特定させる

- Reasonのみ検出精度が低かったため、以降の分析ではReasonは取り扱わない

RQ2: データ加工の結果

要素	含有率
Purpose（目的）	54.54%
Code Explanation（コード説明）	55.39%
Link（リンク）	45.16%
Feedback Type（フィードバック種別）	16.20%
Order（レビュー順序）	6.10%
Test Explanation（テスト説明）	13.66%

RQ2: データ加工の結果

コード変更の目的、コードの説明といった**変更内容を説明する要素**が共通して使用されている

要素	含有率
Purpose（目的）	54.54%
Code Explanation（コード説明）	55.39%
Link（リンク）	45.16%
Feedback Type（フィードバック種別）	16.20%
Order（レビュー順序）	6.10%
Test Explanation（テスト説明）	13.66%

RQ2: データ加工の結果

フィードバック形式の要求やレビュー順番の提案といった**査読者の指示**
に**焦点を当てた要素**は使用されづらい

要素		含有率
Purpose（目的）		54.54%
Code Explanation（コード説明）		55.39%
Link（リンク）		45.16%
Feedback Type（フィードバック種別）		16.20%
Order（レビュー順序）		6.10%
Test Explanation（テスト説明）		13.66%

RQ2:分析

各記述要素の存在がコードレビュープロセスに影響を与えているかを分析

- 分析手法：ロジスティック混合回帰分析
- 詳細は次ページ以降の補足に記載

“ロジスティック混合回帰分析”がよくわからない

➤ 次ページの表はとりあえず以下のように解釈すればいい

- $OR > 1 \rightarrow$ その要素があるとその結果が起きやすくなる
- $OR < 1 \rightarrow$ その要素があるとその結果が起きにくくなる
- $OR = 1 \rightarrow$ 効果なし

(補足) 回帰のパラメータ

使用モデル: 階層ロジスティック混合効果回帰 (lme4
パッケージのglmer関数)

従属変数: マージ決定 (Merge Decision) — 0/1のバイナリ

効果量表現: オッズ比 ($OR = \exp(\beta)$) で報告

- $OR > 1 \rightarrow$ マージされやすくなる
- $OR < 1 \rightarrow$ マージされにくくなる

有意水準: Holm補正済みp値で評価

説明変数: Reason以外のPR説明文記述要素

(補足)交絡因子

| 分析対象ごとに異なる

- マージ決定：貢献者-レビュアー間の過去交流，プロジェクト年齢，コアメンバーかどうか
- Reopen：議論の長さ，コミット数，貢献者のポジティブ感情

RQ2:結果

アウトカム	モデル	Purpose	Code Exp.	Link	Feedback Type	Order	Test Exp.
マージ決定	交絡変数なし	0.978	1.123	1.253	1.645	0.573	1.072
	交絡変数あり	1.010	1.206	1.091	1.721	0.351	1.009
Reopen	交絡変数なし	1.143	0.934	1.047	0.917	0.540	0.999
	交絡変数あり	1.07	0.867	0.958	0.842	0.305	0.923

RQ2:結果

コードの説明がマージの可否に関連してる

アウトカム	モデル	Purpose	Code Exp.	Link	Feedback Type	Order	Test Exp.
マージ決定	交絡変数なし	0.978	1.123	1.253	1.645	0.573	1.072
	交絡変数あり	1.010	1.206	1.091	1.721	0.351	1.009
Reopen	交絡変数なし	1.143	0.934	1.047	0.917	0.540	0.999
	交絡変数あり	1.07	0.867	0.958	0.842	0.305	0.923

RQ2:結果

フィードバックの形式要求を記述するとマージされやすくなる

アウトカム	モデル	Purpose	Code Exp.	Link	Feedback Type	Order	Test Exp.
マージ決定	交絡変数なし	0.978	1.123	1.253	1.645	0.573	1.072
	交絡変数あり	1.010	1.206	1.091	1.721	0.351	1.009
Reopen	交絡変数なし	1.143	0.934	1.047	0.917	0.540	0.999
	交絡変数あり	1.07	0.867	0.958	0.842	0.305	0.923

RQ2:結果

記述要素の有無とReopenされるか否かには関連はない

アウトカム	モデル	Purpose	Code Exp.	Link	Feedback Type	Order	Test Exp.
マージ決定	交絡変数なし	0.978	1.123	1.253	1.645	0.573	1.072
	交絡変数あり	1.010	1.206	1.091	1.721	0.351	1.009
Reopen	交絡変数なし	1.143	0.934	1.047	0.917	0.540	0.999
	交絡変数あり	1.07	0.867	0.958	0.842	0.305	0.923

RQ3:調査対象

データの収集元

- オンラインの開発者コミュニティに所属する開発者

データの収集法

- アンケートを複数のオンライン開発者コミュニティに配布
 - オンライン開発者コミュニティ : Reddit, LinkedIn等

結果

- 57の回答

RQ3:アンケートの項目

1. 参加者の職業的背景（経験年数・役割等）
2. PR説明の遭遇頻度と品質の自己評価
3. 「良いPR説明とは何か」を自由記述で定義
4. 「あなたがレビューしたPRにおいて、説明文の存在はどれほど重要でしたか？」＋その理由
5. 各記述要素について「どのくらいの割合のPRでその要素が重要でしたか？」

RQ3:結果

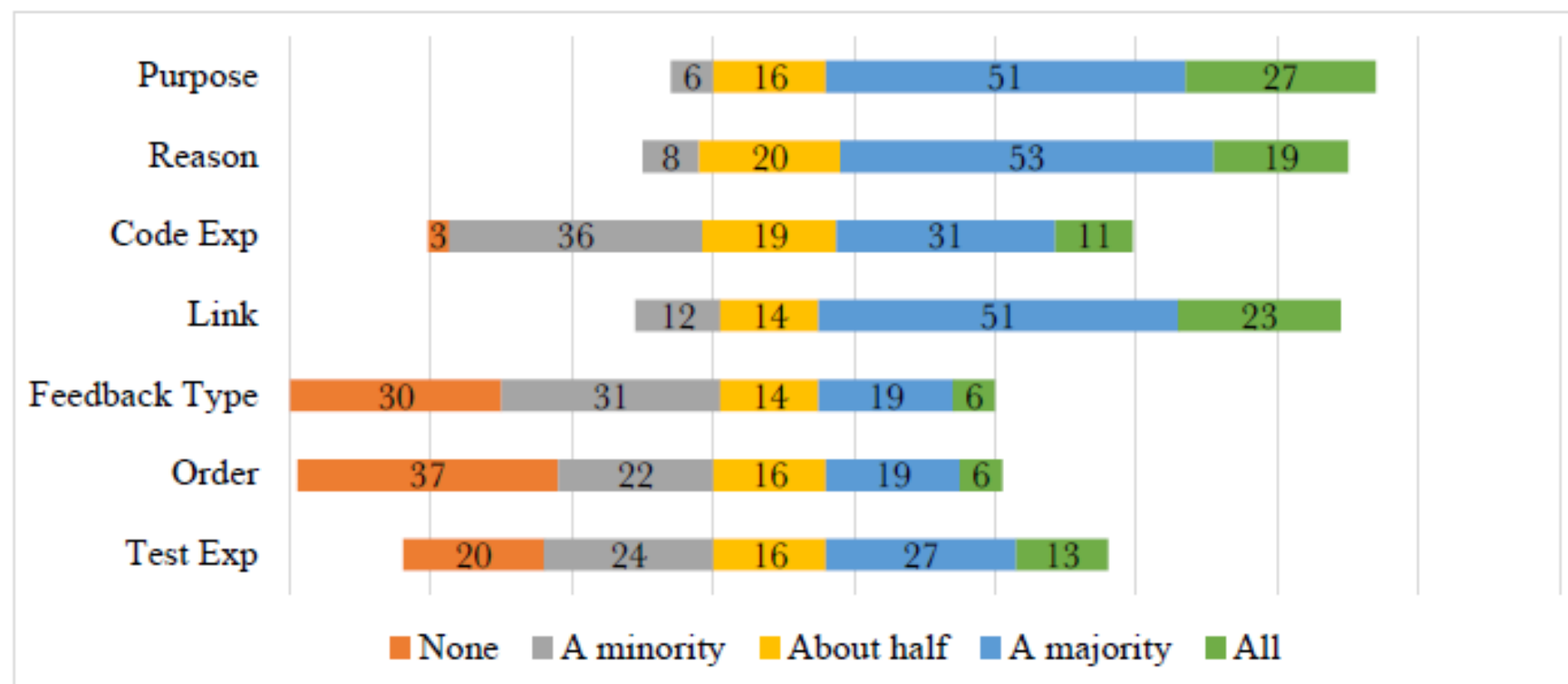
| 開発者にとってのPR説明文の重要度

- 57件中60%が「非常に重要」または「極めて重要」と評価
- 「やや重要」19%、「少し重要」19%、「全く重要でない」3%

➤ 大多数の開発者はPRの説明文を重要視している

RQ3:結果

図：開発者がPR説明文の各記述要素を重要視している割合



Purpose・Reason・Link → 大多数または全PRで重要とされる（最も安定して高評価）

Code Explanation・Test Explanation → 約半数のPRで重要（中程度）

Feedback Type・Order → 重要と感じるPRが少数派（最も低評価）

考察

RQ1で定義した記述要素は大きく2つに分けられる

- 記述的要素 : Purpose, Code Exp, Link, Test Exp
- 相互作用的要素 : Feedback Type, Order

相互作用的要素

- 開発者は重要視していない
- コードレビューの結果に与える影響 : 大

➤ 開発者はPRの説明文を書く際に、積極的に相互作用的要素を記述すべき

結論

研究のアプローチ

- グレー文献レビュー・大規模リポジトリ分析（80K PR / 156 プロジェクト）・開発者サーベイを組み合わせた混合法で、PR説明の価値を多角的に検証した

発見1：PRの説明文の記述要素は2つに分けられる

- 記述的要素：コード変更の理解を支援するものとして開発者に一貫して重要視されている
- 相互作用的要素：使用頻度は低いものの、マージ確率・レビューア関与度において最も一貫して正の結果と結びついている

発見2：PR説明の捉え方を更新すべき

- PR説明は「コード変更を説明するもの」だけではない
- 「レビューアの注意・期待・相互作用パターンを形成するツール」として捉え直すべきである

所感

背景で提起した課題への解答はRQ1, RQ2ですすでに得られている

- RQ1はともかく， RQ2の結果は渋い．．．

個人的には， RQ2で統計的に得られた結果とRQ3の開発の主観から得られた結果の差異が印象的だった

- 記述的要素について， 人間が重要と思ってるなら， コードレビュープロセスにも影響が出そうなのに， なぜか出ていない