

OSS プロジェクト開発者の貢献に関する基礎的分析

Analysis of contributions of developers of open source software project

三谷 幸平[†] 高橋 亨輔[‡] 井面 仁志[‡]
Kohei Mitani Kyosuke Takahashi Hitoshi Inomo

1. はじめに

近年、オープンソースソフトウェア（以下、OSS）開発が活発に行われる中、GitHub などが普及し、開発履歴やプロジェクト内での議論が詳細に蓄積されている。OSS プロジェクト開発に関するデータを分析する研究は盛んに行われており、例として、プロジェクトに参加するコミット数の多い開発者の活動を分析し、特徴を分類する研究[1]や、開発者の中からコミッター（コミット権限を持つ開発者）となる開発者を予測する研究[2]が行われている。蓄積された開発履歴を用いて、開発者の貢献の分類・分析を行うことにより、OSS プロジェクトに参加する開発者の特徴を明らかにすることにつながると考えられる。

一方、OSS では、開発者が新規参入することは困難であることが指摘されている。Steinmacher ら[3]は、新規参入者が OSS プロジェクトに参入する際の最初の貢献を妨げた社会的障壁について分析している。Steinmacher ら[3]は、新規参入者や経験豊富な開発者からの回答から 13 の社会的障壁を含む計 58 の障壁からなるモデルを形成している。この中で、新規参入者は、適切な指導者を見つけることは難しく、OSS への貢献の方法自体が分からないという障壁があげられている。また、OSS を導入する企業も増加している中で開発者不足は問題視されており、新規参入者が OSS のプロジェクトへ参入する支援が必要とされている。

そこで、本研究では、OSS ですでに活動している開発者の貢献の種類や特徴を明らかにすることで、OSS 新規参入者がプロジェクトに参入し、長期的に貢献できるような支援につなげる要素を検討する。具体的には、開発者から注目を集めるプロジェクトとそうでないプロジェクトにおいて、開発者が新規参入した際に初めて行った貢献の種類や、長期的に活動する開発者の変更の特徴を明らかにする。これにより、OSS 新規参入者に対しての OSS プロジェクトにおける貢献の指標を提示し支援を行うことを目指す。

2. 研究概要

本研究の分析対象は、GitHub に登録されているプロジェクトとする。本研究では、開発者が OSS プロジェクトに参入した際の最初の貢献について分析・分類する。開発者の最初の貢献を分析することは、他の開発者がプロジェクトに新規参入する際のきっかけの提示につながると考えられる。一般に、開発者がプロジェクトに対して行う貢献は、コードの変更、課題や問題点の投稿や議論への参加など多様であるが、本研究では、ソースコードやドキュメント変更を行うコミットに限定し分析する。本研究では、以下 3 点に着目して分析する。

- (1) コミットメッセージから読み取れるコミットの意図
- (2) 変更ファイルの形式

[†] 香川大学大学院工学研究科 Kagawa University Graduate School of Engineering

[‡] 香川大学創造工学部 Kagawa University Faculty of Engineering and Design

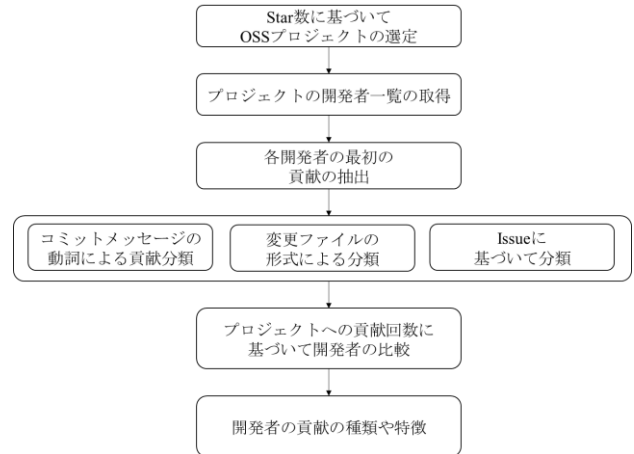


図 1 研究の流れ

表 1 コミットメッセージに頻出する動詞

動詞	概要
add	機能やファイルを追加した際に広く使用
fix	小さな打ち間違えやバグ等を修正した際に広く使用
update	ドキュメント、テスト等の更新した際に使用
change	別のものに変更した際に使用
remove	削除した際に使用
make	対応させたり改良したりした際に広義で使用
use	実装の切り替えや何かを使う際に使用

(3) コミットと課題管理機能「Issue」の関連

まず、(1)の分析では、開発者の貢献の種類を調査する。次に、コミットには、一度のコミットで大量に変更する開発者や、1行だけの変更を加える開発者も存在する。

(2)の分析では、新規参入者は、最初は小さい貢献から始めるのではないかと仮定し、コミットの変更ファイルの形式を調査する。最後に、OSS プロジェクトに投稿される「Issue」は、その時点で必要とされている貢献が具体的に示される。「Issue」は、新規参入者が着目すべきだと考えられるため(3)を調査する。

本研究の流れを図 1 に示す。分類の流れは、以下の Step 1 から Step 5 の通りである。

Step 1 プロジェクトの選定：GitHub で得られるプロジェクトの star 数によって選定を行う。star は開発者が注目したプロジェクトにつけるものであり、プロジェクトの注目度を示す指標となる。star 数が 1000 以上のプロジェクトを注目されているプロジェクトとし、そうでないプロジェクトを 500 付近と 100 未満の 3 つのグループに分ける。

Step 2 開発者の貢献の抽出：Git の log コマンドを用いて、全ての開発者の最初のコミットを取得する。コミットメッセージは、例えば、「Fix mac build」などのように、開発者のコミットの意図を示す文章である。

Step 3 動詞による分類：コミットメッセージの最初の動詞によって貢献を分類する。コミットメッセージは表 1 に示すような動詞で始まるものが多く、コミットメッセージの先頭の動詞により分類する。その他、表 1 に示した動詞以外で始まるものを「Other」、コミットメッセージからは意図が読み取れないものを「Unknown」として分類する。

Step 4 ファイルの形式による分類：開発者が最初に行った貢献で変更されているファイルの形式について注目する。変更されたファイルがソースコードであるか、ソースコードとソースコード以外を同時に変更しているか、ソースコード以外のみであるかを分類する。

Step 5 コミットと「Issue」の関連の有無による分類：コミットが課題管理機能「Issue」に関連するかどうか分類をする。

3. 貢献の分類・分析

本研究では、「apache」から選定した 3 つのプロジェクト（「apache/nifi」、「apache/metron」、「apache/crunch」）を分析対象とする。ここで、「apache/nifi」の star 数は 1,293、「apache/metron」の star 数は 587、「apache/crunch」の star 数は 93 である（2019 年 2 月時点）。

コミットメッセージの動詞による分類を図 2 に示す。図の横軸はプロジェクトごとに総コミット数が「1 回」、「2～9 回」、「10 回以上」で開発者を分けている。縦軸はコミットメッセージの分類の割合を示している。図 2 より、3 つのプロジェクトにおいて、「add」や「fix」に分類されるコミットが多いことが読み取れる。特に、「fix」は 1 行程度のバグ等の修正を多く含むため、新規参加者は軽微な箇所からプロジェクトに参加していると考えられる。

次に、変更ファイルの形式による分類を図 3 に示す。

「java」はソースコードのみの変更を示す。「variety」はソースコードとソースコード以外のファイルを同時に変更している。この中には、一度に多くの変更を行うビッグコミットが含まれていた。「other」はソースコードを含まない変更である。総コミット数が 10 回以上の開発者は、最初のコミットが「variety」に分類されることが多い。

「apache/nifi」は開発者の総コミット数によらず、割合はほとんど同じである。「apache/metron」は「java」に分類される変更をしている開発者は総コミット数によらず、非常に少ない割合を示している。「apache/crunch」は「java」に分類される開発者はどの総コミット数によらずでも 60% を超えており、高い割合を示している。

最後に、コミットと「Issue」の関連の有無による分類を図 4 に示す。「Issue close」されている割合は総コミット数によらず半数を超えており、最初の貢献では「Issue」に関連したコミットが多いことが分かる。

4. おわりに

本研究では、世界中で発展を続ける OSS 開発において新規参加者への支援を行うことを将来的に目指すための、開発者の貢献の分類・分析を行った。

コミットメッセージによる分類においては、「add」や「fix」がプロジェクトの注目度や開発者の総コミット数によらず共通して頻出していた。変更ファイルの形式においては、ソースコード以外のみの変更を行う開発者が多いと予想していたが、ソースコードやソースコード以外のファイルも同時に変更している開発者も多数いることが分かっ

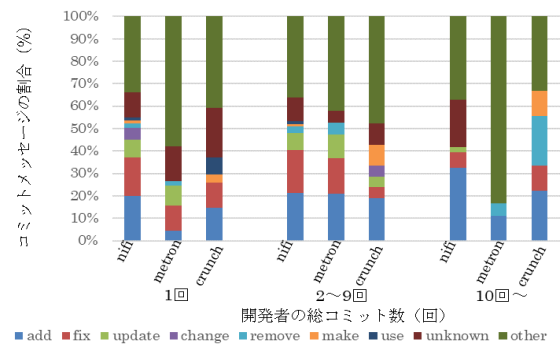


図 2 コミットメッセージの動詞による分類

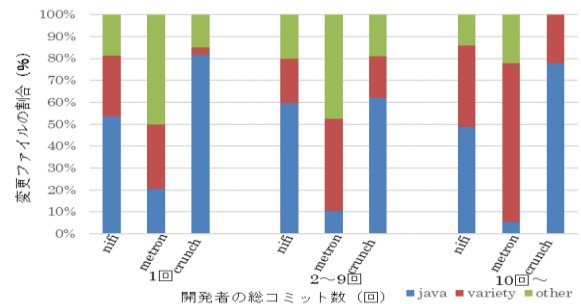


図 3 変更ファイルの形式による分類

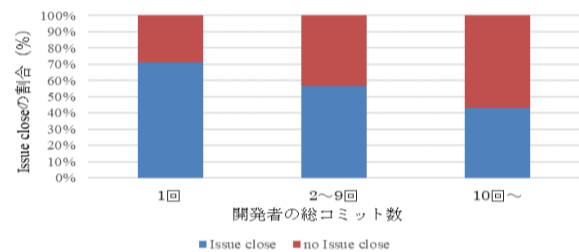


図 4 コミットと「Issue」の関連の有無による分類

た。「Issue」に関しては、開発者の最初のコミットにおける「Issue」との関連は高かった。総コミット数が 10 回以上の開発者の「Issue」との関連が 40% 程度であったが、総コミット数が 10 回以上の開発者はプロジェクトの初期から開発に携わっていることが多く、その段階では、「Issue」が少なかったためであると考えられる。

今後の課題として、より広範囲かつ詳細な開発者の活動履歴の分類・分析、開発者の活動履歴の取得・分類の自動化、開発者に向けた貢献の指標の提示があげられる。

参考文献

- [1] 尾上紗野, 畑秀明, 松本健一, "GitHub 上の活動履歴分析による開発者分類", 情報処理学会論文誌, Vol.56, No.2, pp.715-719, 2015.
- [2] 伊原彰紀, 亀井靖高, 大平雅雄, 松本健一, 鶴林尚靖, "OSS プロジェクトにおける開発者の活動量を用いたコミッター候補者予測", 電子情報通信学論文誌 D, Vol.J95-D, No.2, pp.237-249, 2012.
- [3] Igor Steinmacher, Tayana Conte, David Redmiles, Marco Aurelio Gerosa, "Social Barriers Faced by Newcomers Placing Their First Contribution in Open Source Software Projects", Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing, pp.1379-1392, 2015