

README 修正内容に関連する ソースコード変更コミット追跡への試み

白崎優奈[†]
和歌山大学[†]

伊原彰紀[‡]
和歌山大学[‡]

石岡直樹[§]
和歌山大学[§]

1 はじめに

多くのソフトウェア開発プロジェクトでは、ソフトウェアの機能、使用方法などを開発者や利用者に発信するため、ソフトウェアドキュメントとして README を公開している。ソフトウェア開発の中でも GitHub を利用するオープンソースソフトウェア (OSS) 開発では、開発への参加方法、各機能の概要なども README に記述しており、README はリポジトリのトップページに提示される重要なドキュメントとなっている。

OSS プロジェクトは、実装中のソースコードを逐一公開しているため、ソースコードの更新に伴って README の更新が期待される。しかし従来研究 [1][2] ではソースコードと README が紐付けられていないため、ソースコードと README がどのように共進化しているか明らかでない。また、開発者はソースコードと README を同一コミットで更新していないこともあり、README の更新が必要となったソースコードの変更コミットを紐付けることは容易でない。

本研究は、README 更新の追従性を確認するため、ソースコードと README の変更内容に共通して含まれる文字列に基づいて関連付ける方法を提案する。特に README の変更と、

関連づけられたソースコードのコミットとの時間差を分析することで、プロジェクトにおける README の記述方針を明らかにする。

2 分析

2.1 分析手法

README の変更に関連することが示唆されるソースコードの変更コミットを特定する。具体的には、README の更新で追加・変更された文字列と、各コミットにおいて追加・変更されたソースコード中の文字列をそれぞれ抽出し、同じ文字列を含む README とソースコードを関連づける。ソースコードファイル中から記号削除、NLTK を用いて形態素解析により名詞のみを抽出する。同様の手法で、README 変更コミットで追加・変更された文字列から名詞のみを抽出する。

2.2 ケーススタディ

本論文では、README が英語で記述され、他のリポジトリに比べて頻繁に保守している 2 つのプロジェクト (aliyun-openapi-nodejs-sdk, js-multiaddr) を対象とする。

図 1 と図 2 は、それぞれのプロジェクトを対象に、README の変更コミットの前後 10 コミットそれぞれにおいて、README において変更された文字列を含む頻度を計測した結果である。横軸は README を変更したコミットを 0 とする前後のコミット位置を示し、縦軸はソースコード中に含む README において変更された文字列の数を示す。紙面の都合上、README 記述に追加があった箇所をケーススタディとして示す。

Toward tracking source code change commits related to README revision

[†] Shirasaki Yuna, Wakayama University

[‡] Akinori Ihara, Wakayama University

[§] Naoki Ishioka, Wakayama University

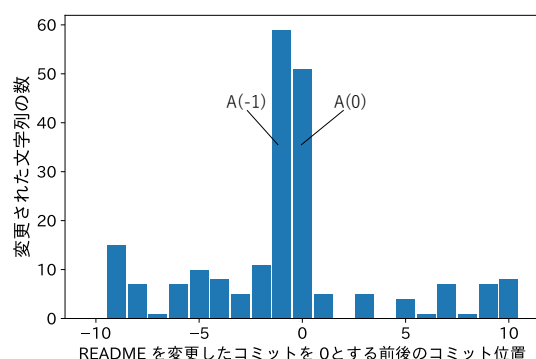


図1: aliyun-openapi-nodejs-sdk における各コミット中の README の変更単語数

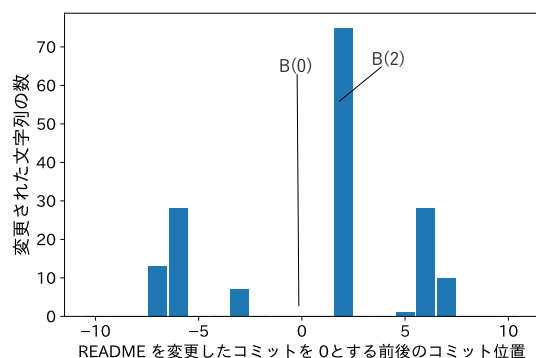


図2: js-multiaddr における各コミット中の README の変更単語数

aliyun-openapi-nodejs-sdk: 図 1 の A(0) において README が更新され, 1 コミット前の A(-1) と A(0) のソースコード変更が README の更新に関連していることが示唆される. 例えば, A(-1) のソースコードでは DescribeAccessPoints によるアクセスポイントに関する記述が追加され, README には, A(0) で DescribeAccessPoints に関する記述が追加されていた. このコミットでは, 実装後に README が更新されたことが示唆される.

js-multiaddr: 図 2 の B(0) において README が更新され, 2 コミット後の B(2) のソースコード変更が README の更新

に関連していることが示唆される. 例えば, B(0) のコミットでは README のみが更新され, プロトコルの項目が細分化されていた. B(2) のコミットでも, プロトコルに関連するファイルの追加やコードの変更が見られた. このコミットでは, 実装前に README が更新されたことが示唆される.

3 おわりに

本論文では, README がソースコードの変更を追従しているかを確認するため, 各ファイルの単語を比較した. 分析の結果, README の更新は, ソースコード変更の前と後のいずれの時期でも変更されることがあり, README の内容に応じて追従するタイミングが異なると考える. 今後は, README とソースコード変更内容による README の追従性違いを明らかにし, README の追従性によるソフトウェア品質への影響を示す. また, README の自動更新タイミングについても検討する.

参考文献

- [1] Gede Artha Prana, Christoph Treude, Ferdian Thung, Thushari Atapattu, and David Lo. 2019. Categorizing the Content of GitHub README Files. *Empirical Software Engineering*. vol.24, no.3, 1296-1327, 2019.
- [2] J. Moreno-Le'on, G. Robles, and M. Rom'an-Gonz'alez, "Dr. scratch: Automatic analysis of scratch projects to assess and foster computational thinking," *RED.Revista de Educaci'on a Distancia*, vol.15, no.46, pp.1-23, 2015.