

グラフモデルを用いた OSS コミュニティ進化構造分析方法の提案と評価

加藤 聖也^{†1} 稲垣 遥太^{†1} 青山 幹雄^{†1}

南山大学 理工学部 ソフトウェア工学科^{†1}

1 研究背景と課題

OSS (Open Source Software) コミュニティ内の開発者ネットワークは複雑化している. OSS コミュニティの動的な構造変化の分析は未確立である. グラフモデルを用いた構造分析のために以下を課題とする.

- (1) OSS コミュニティ構造のグラフモデル表現
- (2) OSS コミュニティ進化の構造分析方法の確立

2 関連研究

- (1) OSS コミュニティとリポジトリマイニング
コミュニティ構造を, 中核, 非中核の開発者に分類するクラスタリングが行われている[1].
- (2) グラフデータベース[3]
グラフデータベース(グラフ DB)は, グラフデータモデルを公開する DB 管理システムである.

3 アプローチ

OSS コミュニティ構造を, グラフ DB を用いてモデル化し, そのクエリを利用して可視化することで OSS コミュニティ構造を分析する.

4 提案方法

提案プロセスを図 1 に示す.

- (1) OSS コミュニティの進化構造の仮説を立てる.
- (2) 仮説に基づいて分析項目を設定する.
- (3) 分析に基づきモデリングを行い, データを収集し, データベースへ挿入する.
- (4) 分析項目に従って分析し, 結果を出力する.
- (5) 分析結果から仮説を検証する. 必要に応じ, 仮説の修正や追加を行い, 一連のプロセスを繰り返す.

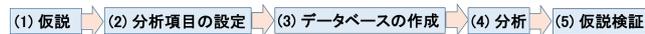


図 1 提案プロセス

5 プロトタイプの実装

図 2 にシステム構成を示す. OSS コミュニティとして GitHub, グラフ DB に Neo4j, グラフの可視化に Neo4j Browser を用いた.

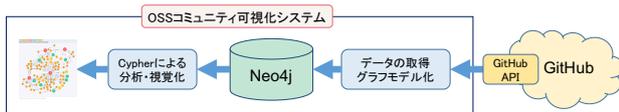


図 2 システム構成図

6 OSS コミュニティへの適用例

6.1 分析対象の OSS コミュニティ

機械学習に関する OSS コミュニティ, TensorFlow, Caffe, Chainer, Jubatus を分析対象とした.

分析において, コミュニティ進化を 3 ヶ月ごとに分割し, それぞれを 1 期, 2 期などとした. また, 3 ヶ月に満たない部分は分析対象外とし, 2011 年 10 月 23 日から 2016 年 12 月 1 日までを分析対象とした.

6.2 分析と仮説検証

6.2.1 3 層で構成されるコミュニティ進化モデル

コミュニティ上の開発者の構造を以下の非中核, 準中核, 中核の 3 層で定義した(図 3).

- (1) 非中核: 中心的役割を果たさなかった開発者
- (2) 準中核: 一定期間のみ中心的に貢献する開発者
- (3) 中核: 長期間, 中心になって貢献している開発者

中核の開発者の構成は, コミュニティ発足時の一部の開発者が主体になっている. この構造はコミュニティが進化しても, 大きく変わらない. 一方, コミュニティ進化に伴い, 準中核の開発者は入れ替わっている. これは新規開発者から準中核の開発者に成長するものが現れるが, 一定期間のみ中心的な役割を果たして, その後はフェードアウトする傾向があると考えられる.

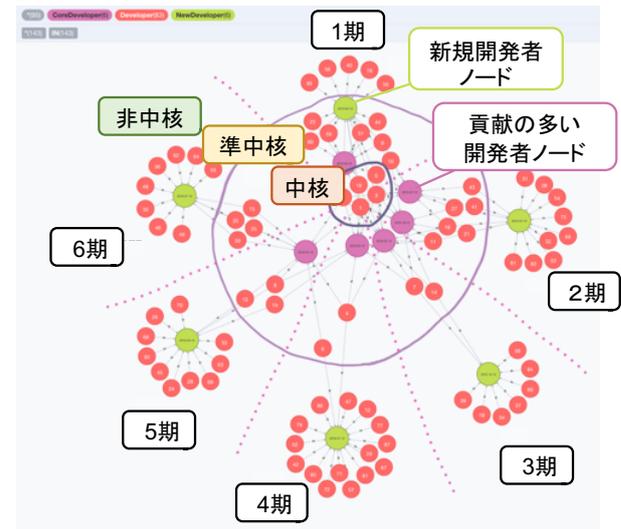


図 3 コミュニティの 3 層構造

6.2.2 開発者の行動による 3 つの成長パターン

新規参入時からデータ取得時までの, 貢献度に差がある開発者 3 人を抽出した. 新規参入時から半年間の行動を分析した.

- (1) 非中核の開発者: 他の開発者とのコミュニケーションは活発であるが, Pull Request の作成は少ない.

A Structural Analysis Method of OSS Community Evolution Based on Semantic Graph Models.

^{†1} Seiya Kato, Yota Inagaki, Mikio Aoyama, Dep. of Software Engineering, Nanzan University

- (2) 準中核的開発者: 他の開発者とのコミュニケーションは少ないが, Pull Request の作成は多い.
- (3) 中核的開発者: 他の開発者とのコミュニケーション, Pull Request の作成が活発に行われている.

また, 3 人の開発者が, コミュニティに貢献したコミット数の累計の推移を分析すると, 開発者の成長にパターンがあることが分かった. 開発者が新規参入してから3期分のデータを使用した(図 4).

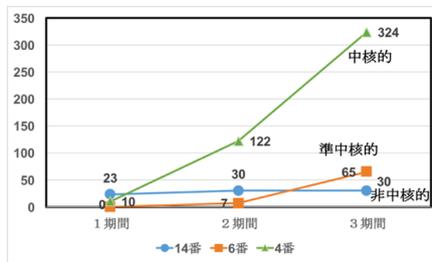


図 4 開発者の成長パターン

6.2.3 中核, 新規開発者のファイル更新による相互作用の推移

リポジトリの成長に伴い, 中核的開発者と新規開発者が共通して更新を行うファイルに, ハイブ曲線のような特徴[2]が表れると考えた. 開発の段階について, 以下の特徴が考えられる(図 5).

- (1) 初期: 新プロダクトへの期待による積極的な更新
- (2) 中期: 新規開発者による機能提案, 実装
- (3) 後期: プロダクトの完成によって更新が減少

また, ファイルの更新時は, ルート直下のドキュメントやセットアップ, コンフィグファイルなどが同時に更新される可能性が高い.

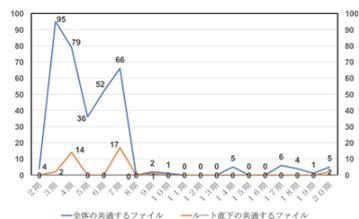


図 5 共通更新ファイル数の推移

7 考察

7.1 グラフモデルによる OSS コミュニティ構造

本稿で提案したプロセスを実際の OSS コミュニティに適用し, 設定した仮説に対して, 適切に分析項目を設定することができた. そのため, OSS コミュニティの動的な構造変化の分析に必要なデータを十分に収集し, グラフモデル表現によるコミュニティのモデリングが可能となった.

7.2 OSS コミュニティの動的な構造変化の分析方法の確立

以下の OSS コミュニティの動的な構造変化を明らかにした.

- (1) 非中核, 準中核, 中核的開発者の 3 層コミュニティ進化モデル
- (2) 開発者の行動による成長パターン
- (3) 開発段階による開発者間の相互作用の変化

7.2.1 コミュニティ構造と進化モデル

関連研究では, OSS コミュニティの構造を, 中核,

非中核的開発者として分類していた[1]. 本稿では, 開発者を中核, 非中核に加えて準中核的開発者に分類が可能であることを明らかにした. これより, コミュニティ上の開発者の役割構造をより明確にすることが可能になった.

進化モデルとして, 中核的開発者を中心に開発が進み, 新規開発者の参入と, その中から準中核的開発者が現れ, それが入れ替わりながら進化していることを明らかにした. これはコミュニティの運用の支援に活用が期待できる.

7.2.2 開発者の行動による成長パターンの特性

開発者の行動パターンにより, 開発者の成長に差が現れることが明らかになった.

行動パターンは, 2つに分類することができる.

- (1) 非中核的開発者: バグ報告や機能提案を中心
 - (2) 準中核, 中核的開発者: 機能の実装を中心
- 開発者の成長パターンを, 3つに分類できる.

- (1) 非中核: 貢献数がほぼ増加せず, 成長率が低い
- (2) 準中核: 貢献数は増加するが, 成長率が低い
- (3) 中核: 貢献数が増加し, 成長率も高い

準中核と中核的開発者の違いは, 他の開発者とのコミュニケーション量の違いであり, 継続的に機能提案, 実装を行う開発者は, 他の開発者とのコミュニケーションも多いと考えられる. これらのパターンは, 開発者の成長支援に活用できる.

7.2.3 開発段階による開発者間の相互作用の変化

開発段階によって, ファイル更新に中核, 新規開発者間の相互作用がみられた. 開発は初期, 中期, 後期の 3 つに定義でき, 開発の初期と中期で双峰性が見られ, ハイブ曲線[2]の波形と似た意味を持つと考えられる.

8 まとめ

OSS コミュニティ構造にグラフモデル表現が適していることを明らかにし, OSS コミュニティ進化の動的な構造変化の分析方法を提案した. 分析により, 以下の 3 つの特性を発見した.

- (1) 3 層コミュニティ進化モデル
- (2) 開発者の行動による成長パターン
- (3) 開発段階による開発者間の相互作用の変化

これらを用いることで, OSS コミュニティの運用や開発者の成長の支援への活用が期待できる.

参考文献

[1] A. Bosu, et al., Impact of Developer Reputation on Code Review Outcomes in OSS Projects, Proc. of ESEM '14, Article No. 33, Sep. 2014.
 [2] J. Fenn, et al., Mastering the Hype Cycle, Harvard Business School Pr. 2008
 [3] I. Robinson, et al., Graph Databases, O'Reilly, 2015.