

機械学習によるグラフモデル OSS 開発コミュニティ構造の 特徴量分析方法の提案と評価

加藤 聖也^{†1} 青山 幹雄^{†1}

南山大学大学院 理工学研究科 ソフトウェア工学専攻^{†1}

1 研究背景と課題

近年, OSS (Open Source Software) 開発コミュニティにおける開発者間ネットワークは複雑化している. OSS 開発コミュニティの動的な構造変化に着目した研究として, OSS 開発コミュニティ構造の動的変化をグラフモデルを用いて定義した研究がある[2]. また機械学習分野において, グラフのリンク予測やノードラベリングに使用可能な汎用性のある特徴量獲得方法が提案されている.

本研究では背景を踏まえ, グラフの特徴量を得る手法を OSS 開発コミュニティデータに適用することで, OSS の大規模な開発データに表れる特性の分析を目的とし, 以下の 2 点を研究課題とする.

- (1) OSS 開発コミュニティの特徴量分析方法の確立
- (2) 実 OSS 開発コミュニティへの適用と有効性評価

2 関連研究

(1) OSS 開発コミュニティと進化

OSS 開発コミュニティ構造を中核, 非中核の開発者への分類[1]や加えて準中核的な構造の発見[2], また, 8 階層の役割変化プロセスを定義した研究がある[4].

(2) グラフモデル

プロパティグラフは, ノード, エッジ, プロパティによって構成されるグラフ型データモデルの一種[5]である. 既存研究[2]ではプロパティグラフを用いて OSS 開発コミュニティ構造のモデル化を行った.

(3) ネットワーク表現学習

ネットワークの表現学習はネットワーク構造からノードの分散表現を獲得する手法である. このような手法として graph2vec[3]などがある.

3 アプローチ

OSS 開発コミュニティを理解するためには大規模な開発データを分析する必要がある. また, 開発者は関連性のある OSS についても貢献活動を行なっている可能性があり, OSS の進退を理解するためには OSS 間のマクロな視点での分析が必要と考える. 既存研究[2]では, OSS 開発コミュニティを定義した新しいクラスのグラフモデルである SCGM (Software Community Graph Model) を提案した. 図 1 に定義したグラフモデルを示す. 本研究では, このモデルを用いて表現した OSS 開発コミュニティ構造から特徴量を抽出し, 分析を行う.

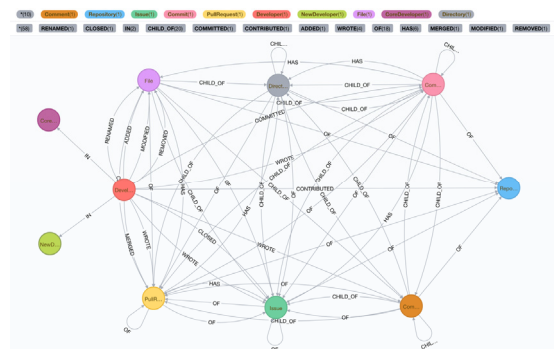


図 1 SCGM (Software Community Graph Model) の定義

4 提案方法

提案する特徴量抽出プロセスを図 2 に示す.

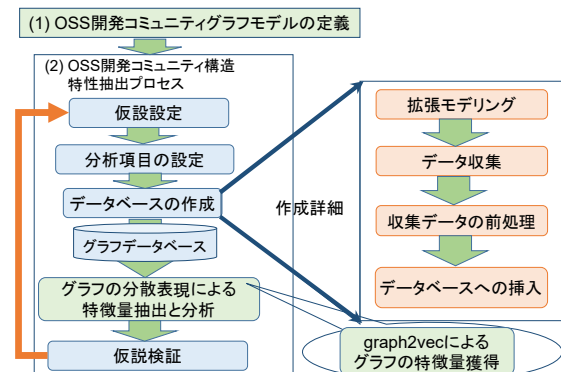


図 2 特徴量抽出プロセス

特徴量抽出プロセスは以下の 2 つのステージからなる.

- (1) OSS 開発コミュニティグラフモデルの定義
対象とする OSS 開発コミュニティを基にグラフモデルを定義する. 本稿では, GitHub 上のデータを利用して定義した SCGM を用いる.
- (2) OSS 開発コミュニティ構造特性抽出プロセス
 - (a) OSS 開発コミュニティの進化構造の仮説を立てる.
 - (b) 仮説に基づいて分析項目を設定する.
 - (c) グラフ DB 上で SCGM インスタンスの作成
 - (d) グラフの分散表現による特徴量抽出と分析
 - (e) 分析結果をもと仮説検証を行う.
 必要に応じ, 仮説の修正や追加を行い, 上記(2)の一連のプロセスを繰り返す.

5 SCGM 上の機械学習による OSS 開発コミュニティ分析方法

5.1 システム構成

表現学習を用いた OSS 開発コミュニティの特徴量分

A Feature Analysis Method with Machine Learning of OSS Development Community Structure Based on Graph Models.

^{†1} Seiya Kato, Mikio Aoyama, Graduate Program of Software Engineering, Graduate School of Science and Engineering Nanzan University

析システムの構成図を図3に示す。Graph2vecによる特徴量抽出プロセスでは、機械学習フレームワークのTensorFlow, クラスタリングおよび分析には scikit-learn を使用した。

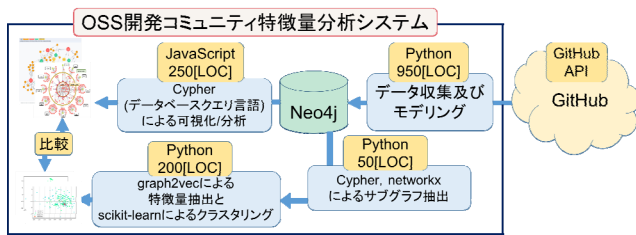


図3 システム構成図

5.2 実 OSS 開発コミュニティへの適用

提案方法による特徴量分析の有効性を評価するため、実際の OSS 開発コミュニティとして Caffe, Caffe2, Chainer, Keras, PyTorch, Theano, Apache Hadoop に対して適用した。100 エポック学習させた結果得られた特徴量を基に、k-means 法を用いてクラスタリングを行った。

5.3 分析結果

クラスタリングの結果および、エルボー曲線を図4に示す。今回、クラスタの数は3としている。クラスタリングでは、丸、四角、三角に振り分けている。

(1) 開発者の3層構造の特定

図4左が k-means 法によるクラスタリングであり、図4中央は主成分分析によって次元削減を行なった後の k-means 法によるクラスタリングの結果である。次元削減を行わない場合では、グラフ上に丸、四角が幅広く分布しており、各開発者のクラスタが判別しにくい。一方、次元削減を行なった場合では、特に Chainer や Keras, Theano など少数の開発者が密集しているクラスタが顕著に見られる。これが、中核の開発者であると考えられる。また、多数の開発者が幅広く分布しているクラスタと、そこから徐々に離れている開発者が存在する。これらが、3層構造における非中核の開発者と準中核の開発者と考えられる。

(2) クラスタ数の妥当性

各コミュニティのエルボー曲線を図4右に示す。Chainer, PyTorch ではクラスタ数3以降から誤差が減っている。また keras では、クラスタ数4から5で誤差が上昇している。その他のコミュニティではクラスタ数10まで誤差が大きく変わる箇所が少なかった。全体としては、エルボー法ではクラスタ数の決定は難しいと言える。また、scikit-learn のシルエット解析を用いてスコアを算出した結果、同様に有効な値を得ることは難しかった。

6 考察

既存手法では、OSS 開発コミュニティの開発者の3層構造は力学シミュレーションの結果から目視で判断していた。本稿では、開発者の行動グラフから抽出した特徴量から2層以上の構造に分類が可能であるといえ、機械的な判断でも準中核の開発者の存在が示唆できると言える。また、機械学習ライブラリの OSS と属性が異なる Apache Hadoop に対しても適用したが、特徴量は似

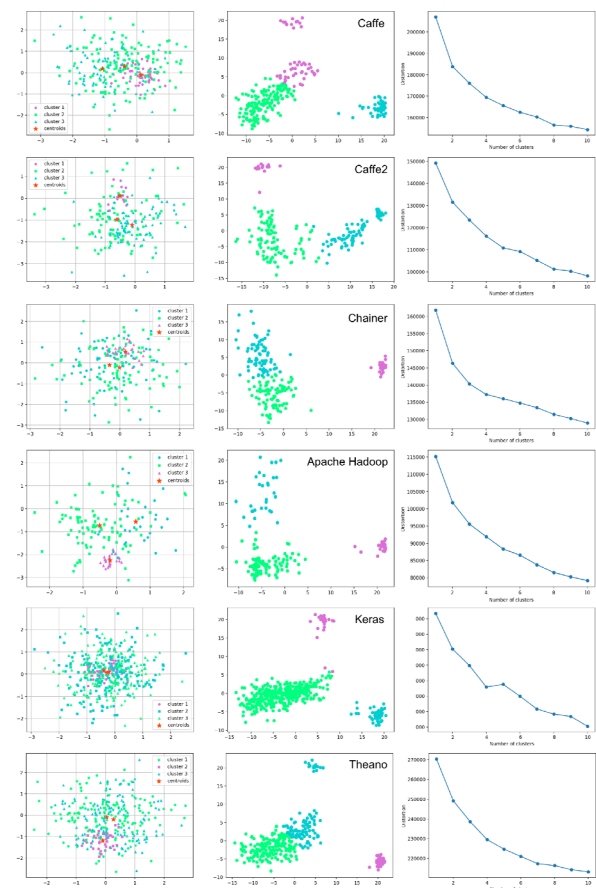


図4 クラスタリング結果

た結果を示し、提案手法が機械学習以外の OSS へも適用できる可能性が示唆される。

7 今後の課題

本稿では、表現学習によって獲得した特徴量を持つ開発者行動の意味分析を行っていない。そのため、どのような開発者行動が特徴量に影響を与えるかを検証する必要がある。

8 まとめ

既存研究では、OSS 開発コミュニティの3層構造は生成した可視化グラフから発見されるものであった。本稿では、ネットワーク表現学習を用いて開発者の行動グラフから特徴量を獲得し、クラスタリングを行うことでコミュニティ構造を把握する手法を提案した。実際の OSS 開発コミュニティデータに対して適用し、有効性を示した。

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, ACM, Sep. 2014, 10 pages.
- [2] S. Kato, et al., A Structural Analysis Method of OSS Development Community Evolution Based on a Semantic Graph Model, Proc. of COMPSAC 2018, IEEE, Jul. 2018, pp. 292-297.
- [3] A. Narayanan, et al., graph2vec: Learning Distributed Representations of Graphs, Proc. of MLG Aug. 2017, 8 pages, <http://www.mlgworkshop.org/2017/>.
- [4] K. Nakakoji, et al., Evolution Patterns of Open-Source Software Systems and Communities, Proc. of IWPSE '02, ACM, May 2002, pp. 76-85.
- [5] I. Robinson, et al., Graph Databases, 2nd ed., O'Reilly, 2015.