

Classifying Developers into Core and Peripheral: An Empirical Study on Count and Network Metrics



**Mitchell Joblin, Sven Apel, Claus Hunsen,
Wolfgang Maurer**

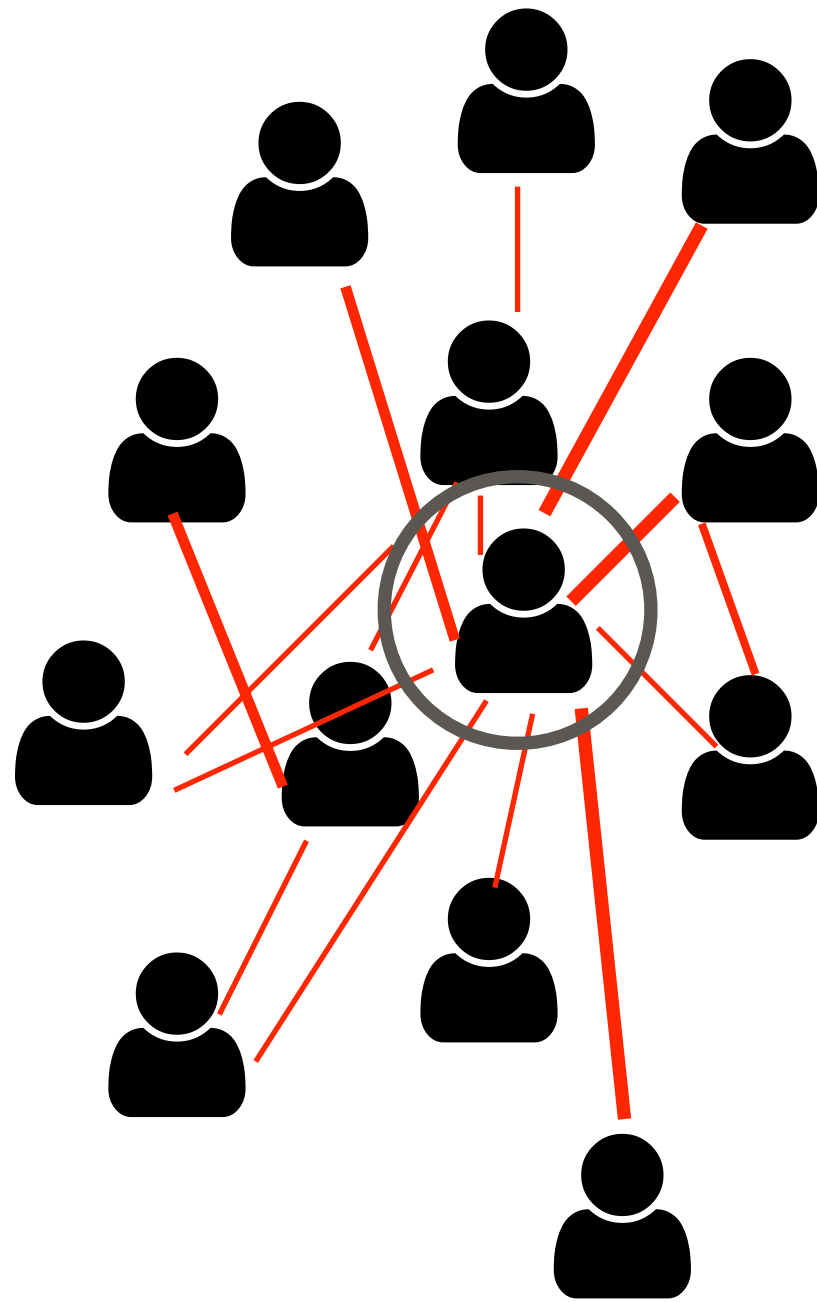
ICSE '17 Proceedings of the 39th International Conference on Software Engineering
Pages 164-174

紹介者：福井 克法

背景

- ◎ ソフトウェアプロジェクトでの開発者の役割を知ること
は、プロジェクトを理解する上で重要
- ◎ 開発者は、コア開発者とペリフェラル開発者で分類される
場合が多い
 - ▶ カウントベースの分類(コミット数, 変更コードの行数,
送信されたメールの件数)
 - ▶ ネットワークベースの分類(他の開発者との共同作業)

ネットワークベースの手法



- ◎ 他の開発者とコミュニケーションをとっている開発者をコア開発者とみなす手法
 - ▶ メーリングリストのやりとり
 - ▶ ソースコード編集の共同作業
- ◎ 階層的クラスタリング
- ◎ 分割最適化クラスタリング

問題

- 開発者をコア開発者とペリフェラル開発者に分類する手法は、カウントベースの手法、ネットワークベースの手法どちらも数多く存在する
- しかし、手法の妥当性を示す根拠が示されているものはない
- カウントベースの手法、ネットワークベースの手法どちらが精度が良いかも検証されていない

目的

1. カウントベースの手法とネットワークベースの手法にどの程度の妥当性があるかを示すこと
2. カウントベースの手法とネットワークベースの手法では、どちらが優れているかを比較する

リサーチクエスチョン

◎ RQ1

- ▶ カウントベースの手法は、どの程度正確であるか？

◎ RQ2

- ▶ ネットワークベースの手法はどの程度正確であるか？

◎ RQ3

- ▶ カウントベースとネットワークベースでの精度の差はあるのか？

評価方法

- 10件の調査プロジェクトで、開発者に対してアンケートを行う
- 169人の回答者のアンケート結果をカウントベースの手法、ネットワークベースの手法の精度を図る

アンケートの内容

自分の役割を自己報告
(コアまたはペリフェラル)

12人の開発者をコアか、そうでないかに分類

カウントベースの
手法で分類された
コア開発者 5名

コア開発者の周辺
に存在する開発者
5名

回答者の周辺に存
在する開発者 2名



コア開発者かペリフェラル開発者か判断する

対象プロジェクト

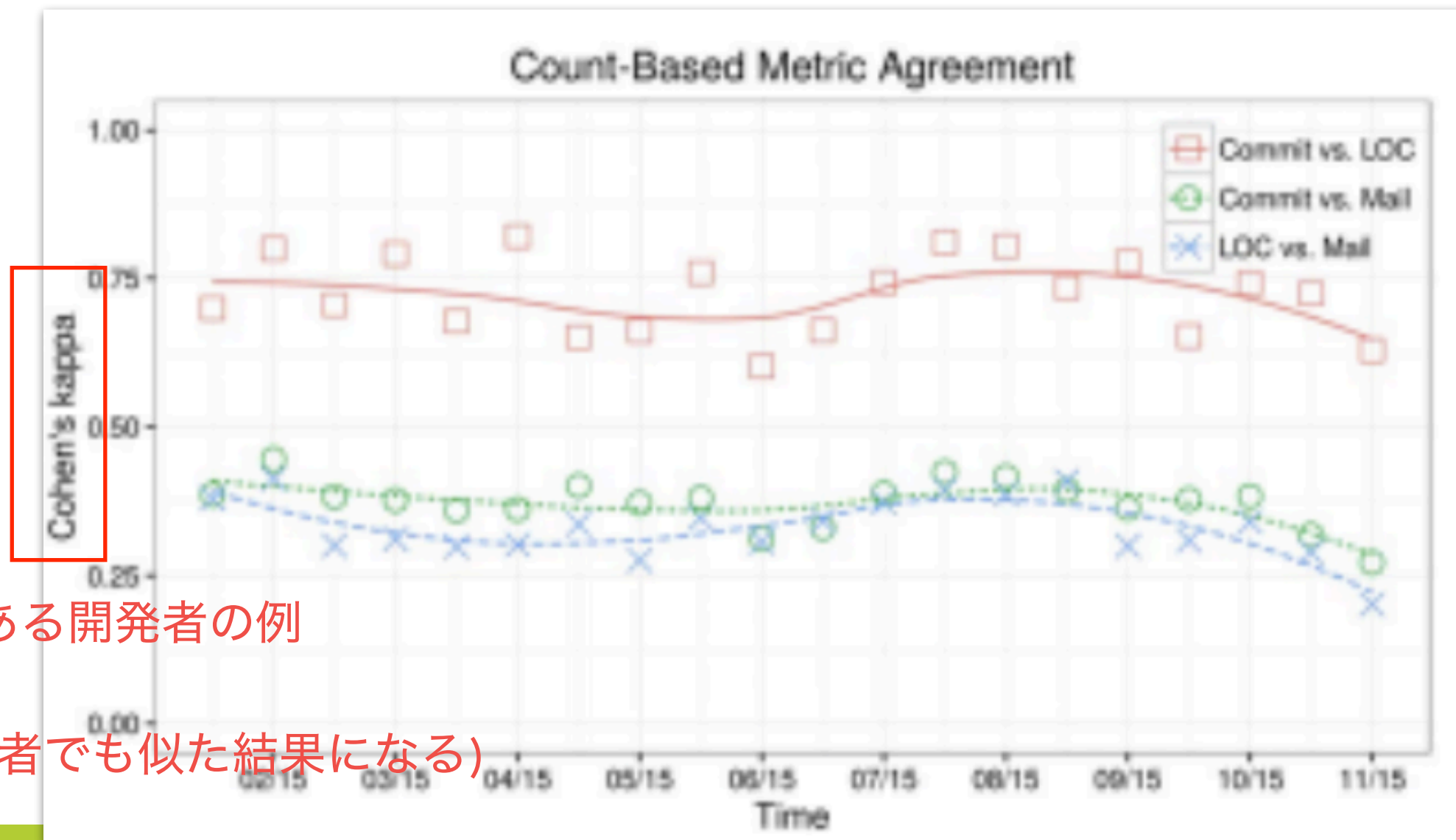
◎ 10プロジェクトを対象とする

Project	開発者数	ソースコード長	コミット数
Project X	826	10M	276K
Django	100	430K	41K
FFmpeg	103	1M	78K
GCC	122	7.5M	144K
Linux	1467	18M	637K
LLVM	180	1.1M	62K
PostgreSQL	17	1M	40K
QEMU	134	1M	43K
U-Boot	142	1.3M	35K
Wine	62	2.8M	110K

結果：RQ1

● RQ1: カウントベースの手法は、どの程度正確であるか？

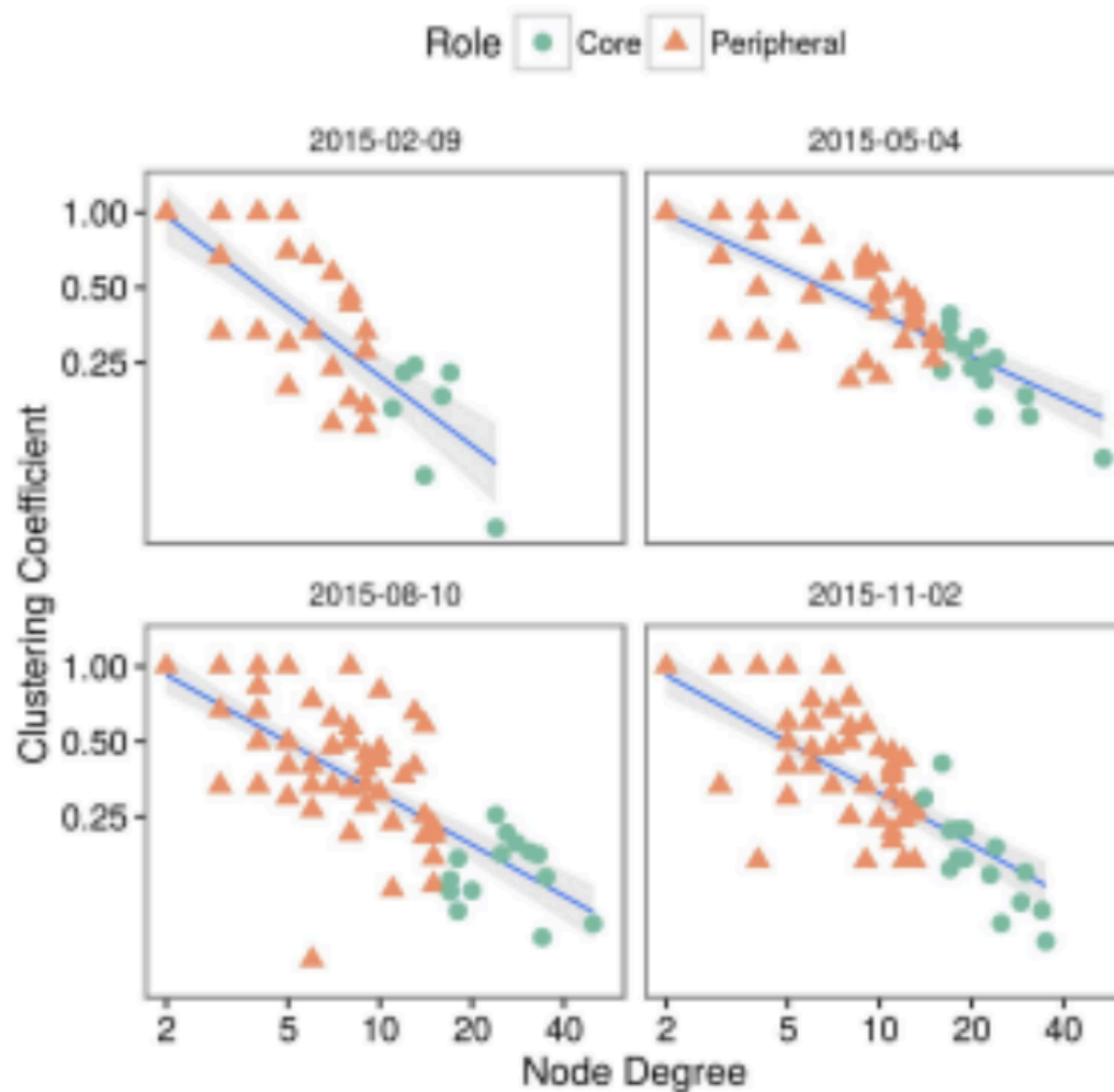
- ▶ 各メトリクスでコア開発者とペリフェラル開発者を分類した際のアンケート結果との一致度の比較



ある開発者の例

(どの開発者でも似た結果になる)

結果：RQ2



- 図はスピアマンの相関係数を用いた分布
- コア開発者とペリフェラル開発者を次数(繋がっている開発者数)に応じてクラスタリングする
- 次数が高いほど, コア開発者になる

結果：RQ3

● RQ3：カウントベースとネットワークベースでの精度の差はあるのか？

● アンケート結果と、分類精度の比較を行った

▶ ある開発者を例にしたアンケートとの一致度

ベース手法		開発者の評価との一致度
カウントベース手法	Commit Count	0.387
	LOC Count	0.355
	Mail Count	0.421
ネットワークベース手法	VCS Degree	0.465
	VCS Hierarchy	0.437
	VCS EigenCent	0.404
	Mail Degree	0.497
	Mail EigenCent	0.427

まとめ

- ◎ コア開発者とペリフェラル開発者の分類において, カウントベースの手法とネットワークベースの手法を比較した
 - ▶ ネットワークベースの手法の方が精度が高い
- ◎ ネットワークベースの手法を用いることで, カウントベースの手法では考慮されていなかったコア開発者を分類することが可能になった

所感

- 特定の分野での主要開発者（コア開発者）を特定するには、ネットワークベースの手法が有効であることは自分の研究でも使えそうだと感じた
 - ▶ カウントベースでは考慮されていなかった開発者を含めることができる点で
- ネットワーク分析に関する知識が乏しいため、勉強する必要があると感じた