

git リポジトリから工数を算出する手法の支援についての検討

戸 田 航 史^{†1}

最近になり、OSS プロジェクトにおける特定期間内に投入された工数をそのリポジトリのデータから算出する方法が提案された。この方法は開発者を一定期間内のコミット数によりフルタイム開発者とそれ以外に分割することで成立している。しかしながら、フルタイム開発者とそれ以外を分けるコミット数（閾値）は当該論文では各開発者へのメールによるアンケートを基に決定されており自動化が困難である。本稿ではその閾値を ISBSG のデータを利用して算出する方法について検討する。

A supporting method for effort calculation from git repository

KOJI TODA ^{†1}

Effort calculation method using VCS system is suggested in previous study. In the method, developers are divided to fulltime developers and non fulltime developers by a number of commit between releases. However, a threshold of full time and non-full time developer is decided by questionnaire to developers. In this paper, I discuss automatically threshold decision method using ISBSG dataset.

1. はじめに

ソフトウェア開発において、開発工数の見積もりは人的資源やスケジュール管理の基礎となるものであり、できるだけ高い精度で見積もる必要がある。そのためこれまで様々な見積もり方法が提案され、その性能の評価、比較が行われてきた。これまで、工数見積もり手法で対象とする「工数」とは「開発者の人数 × 開発期間」で算出される値であった。ただし、この算出方法には企業が受注開発で作成するソフトウェアでは利用できるが OSS では利用できないという問題があった。これは、前述の式には多少の例外はあるものの、おおむね「全ての開発者は等しく毎日一定の時間を開発に費やす」という前提に基づくものであるのに対し、OSS プロジェクトでは開発者の多くはボランティアであり、プロジェクトへの貢献の頻度も貢献の大きさも開発者ごとに大きく異なる。このため、OSS のプロジェクトには既存の工数算出式が適用できなかった。しかし最近になり、OSS に参画する開発者の貢献の頻度を用いて、以下の手順で工数を算出する方法が提案された¹⁾。

- (1) 期間 T (ヶ月) ごとに各開発者のコミット回数を集計する

- (2) 期間 T 中のコミット回数が閾値 t 以上の開発者をフルタイム開発者、 t 未満をパートタイム開発者に分類する。フルタイム開発者数を n_{ft} 、パートタイム開発者数を n_{pt} と定義し、各パートタイム開発者の期間 T 内でのコミット数を C_{pt} とする

- (3) $(n_{ft} + \sum_1^{n_{pt}} \frac{C_{pt}}{t}) \times T$ で算出される値を工数とする

このうち n_{ft} 、 n_{pt} 、 C_{pt} は計測可能であり T はリリースサイクルであるため既知である。しかし、 t は開発者らへのメールによるアンケートを基に算出されており、当該論文の方法に従う場合、工数を算出したプロジェクトごとに開発者にアンケートを実施する必要があることになり、非常に高コストである。

そこで本研究では、企業で収集されたプロジェクトデータである ISBSG のデータを用いて開発者をフルタイムとそれ以外に分けるための閾値を算出する方法を提案、検証する。

本研究における仮説を以降に述べる。OSS は参加する開発者が存在する限り開発は継続して行われる。これは開発種別としては拡張開発に相当すると考えられる。すなわち、OSS は最初のリリースまでを新規開発、それ以降の開発を拡張開発と見なす事ができる。この場合、最初のリリースからある時点までのソースコード行数の増加と、その間のコミットから算出され

^{†1} 福岡工業大学

Fukuoka Institute of Technolog

る工数は、本研究では ISBSG データ中の拡張開発プロジェクトのソースコード行数と実装工数に対応づける事が可能となる。この対応付けから、前述のコミット数を用いた工数の算出における t が決定可能と考えられる。

2. 実験

実験では OpenStack^{*1} の git リポジトリと ISBSG D&E Repository²⁾ を用いた。

OpenStack はクラウドコンピューティングのインフラを構築、運用するためのソフトウェア群、ISBSG は International Software Benchmarking Standard Group (ISBSG) が収集したプロジェクトデータである。今回は 1989 年から 2012 年の間に世界 21ヶ国のソフトウェア開発企業から収集されたデータを用いる。

OpenStack は 1st リリースから 2010 年 12 月 31 日までの期間を対象とし、StackForge 等の派生プロジェクトは対象にしなかった (git リポジトリの OpenStack ディレクトリ以下のみを対象とした)。OpenStack のデータ選定条件は文献 1) に準じた。ISBSG の条件は以下の通りである。開発種別が拡張開発 (Development Type: Enhancement) である、実装工数 (Effort build) が記録されている、工程が記録されていない工数 (Effort Unrecorded) が存在しない、工数が開発者ごとに毎日記録されている (Recording Method: Staff hours(recorded))、工数計測の対象者が開発者のみ (サポート要員を含まない。Resource Level: 1)。各条件は OpenStack のデータ選定条件に近づくように設定した。選定の結果、18 プロジェクトがその条件を満たした。

2.1 実験手順

実験手順を以下に示す。OpenStack の git リポジトリから算出される工数については文献 1) で示されている値を用いた。

- (1) git リポジトリを first release 時点 (2010 年 11 月 1 日とした) に巻き戻し、2013 年 12 月 31 日時点との行数の差を算出する
- (2) 選定済みの ISBSG のデータを対象に、目的変数を実装工数、説明変数をソースコード行数とする対数単回帰分析を行う
- (3) (1) で算出した行数の差を (2) で構築したモデルに代入し、工数を算出する
- (4) git リポジトリから算出された工数と (3) で算出された工数の比較を行う

*1 <http://www.openstack.org>

期間	SLOC	総行数
1st release	845,832	1,192,792
2013/12/31	2,979,542	3,981,902
差分	2,133,710	2,789,110

2.2 実験結果

OpenStack の first release と 2013 年 12 月 31 日時点でのソースコード行数データを表 1 に、ISBSG データから構築された単回帰モデルを式 1 に示す。

$$\log(\text{実装工数}) = 0.566 \times \log(\text{SLOC}) - 1.505 \quad (1)$$

式 (1) に表 1 で得られた SLOC の値を代入した結果、OpenStack の見積もり工数は 118.9 人月となる事が分かった。しかし、git のリポジトリから算出される工数は工数が最大となる $t = 1$ のとき 17,400、 $t = 20$ のときでも 6,945 であり、少なくともおよそ 70 倍程度の差がある事が明らかとなった。

3. おわりに

本稿では OSS プロジェクトの工数を git のリポジトリから見積もる方法と、ISBSG のデータからソースコード行数-実装工数の対数単回帰モデルから工数を見積もる方法を比較した。その結果、2 者間には大きな乖離がある事が分かった。その原因は、そもそも工数の算出基準が異なっており、そこから生まれる差異が著しい事にあると考えられる。従って、git のリポジトリから算出される工数を、1 開発者の 1ヶ月の開発が 1 人月という既知の工数と同一視できない事が明らかとなった。当初はフルタイム開発者と非フルタイム開発者を分けるコミット数の閾値 t を自動的に定める方法を検討する予定であったが、本稿の結果を踏まえるに、全く異なる方策が必要と考えられる。今後、例えば t の決定方法として、コミット数の多い開発者上位 $n\%$ をフルタイム開発者と見なす等のアプローチを実施する予定である。

参考文献

- 1) Gregorio Robles, JesúsM. González-Barahona, Carlos Cervigón, Andrea Capiluppi, and Daniel Izquierdo-Cortázar. Estimating development effort in free/open source software projects by mining software repositories: A case study of openstack. In *Proceedings of the 11th Working Conference on Mining Software Repositories*, MSR 2014, pp. 222–231, 2014.
- 2) The International Software Benchmarking Standards Group. ISBSG D&E corporate release april 2012, 2012.