

# 計算機上のソフトウェア開発作業の自動推定の検討

清水良介<sup>†1</sup> 門田暁人<sup>†1</sup> ゼイネップ ユジャイ<sup>†1</sup> 上野 秀剛<sup>†2</sup>

**概要**: 本稿では、アプリケーション自動計測ツールを用いた、計算機上のソフトウェア開発作業の自動推定について述べる。

## On Estimating Software Development Tasks in Computer Environment

RYOSUKE SHIMIZU<sup>†1</sup> AKITO MONDEN<sup>†1</sup>  
ZEYNEP YUCEL<sup>†1</sup> HIDETAKE UWANO<sup>†2</sup>

### 1. はじめに

Tom DaMarco の「計測できないものは制御できない」という格言が知られるように、開発プロセスの制御や改善には計測が必須である[1]。開発プロセスで生じる問題の多くは人的要因に起因する[2]ことから、プロダクトやプロセスを計測することに加えて、開発の主体である人間やその作業を直接計測し、改善につなげることが重要であると考えられる。

その一つの手段として Personal Software Process (PSP) / Team Software Process (TSP) [3]が知られている。PSP/TSP では、開発者もしくは開発チームが、日々の開発タスクに従事した時間や作業量を計測する。ただし、その計測は人手で行う必要があることから導入の敷居は高く、広く普及するには至っていない。

そこで上記の開発タスクに関する情報を自動計測するツールとして、ソフトウェア開発行動記録システム TaskPit[5][6]が開発されてきた。TaskPit は「計測に基づくプロセス改善」を実践し世の中に広めていくことを目的とし、計算機上で実行されるアプリケーションとそのウィンドウ名にひも付けられたタスクを自動計測できる。

ただし、このような自動計測ツールで計測されるタスクの種類は、開発作業として記録すべき作業の種類と異なっていることが問題である。例えば、TaskPit では、Word を使った「文書閲覧」、ブラウザによる「ブラウジング」といった、アプリケーションやウィンドウにひも付けられたタスクを自動計測できる。しかし、開発作業として記録すべき作業は、「プログラミング」「仕様書作成」「テスト」といったものであり、これらの作業時間を知ることで、計画通りのプログラミング時間を確保できているかや、計画を超過

してテスト作業が発生しているか等を判断し、プロセス改善につなげることができる。

そこで、本稿では、TaskPit にて計測できるアプリケーションとウィンドウ名の履歴から、ソフトウェア開発に即した作業種別を自動推定することを目的とする。従来、このような開発作業の自動推定を目的として、機械学習を用いる方法が提案されている[4]が、学習データに存在しないアプリケーションの実行系列が出現した場合に推定が行えないという問題があった。また、開発者組織や開発者毎に作業推定モデルを構築する必要があり、学習データに対するラベル付けに多大な時間を要するという問題があった。

そこで、本研究では、アプリケーションとウィンドウ名と、開発作業との対応を、1 対多対応のルールとして記述し、そのルールに基づいて各時刻におけるもっともらしい開発作業を推定する方法について検討する。

### 2. TaskPit のデータ例

図 1 に TaskPit の実行画面の例を示す。また、表 1 に TaskPit によって計測されるデータの例を示す。TaskPit では、計算機上のアプリケーションの実行ファイル名、開始時刻、終了時刻、左クリックの回数、右クリックの回数、打鍵回数、実行ファイル名（およびウィンドウ名）が記録され、実行ファイル名とウィンドウ名に応じてあらかじめ設定されたタスク名が記録される。例えば、表 1 の計測例の 2 行目では、2013 年 5 月 10 日 15 時 22 分 8 秒から 11 秒にかけて `firefox.exe` が実行されており、そのウィンドウ名は、「#886 (trac に upload した設計書の履歴情報について)? Solution - Mozilla Firefox」であった。また、このときに、左クリックが 1 回実行されている。

表 1 では、「TaskPit」「ブラウザ」「ファイル操作」といったタスク名が記録されているが、本稿では、「プログラミング」「テスト」といったソフトウェア開発に即した作業名として推定することを目的とする。

<sup>†1</sup> 岡山大学  
Okayama University

<sup>†1</sup> 奈良工業高等専門学校  
National Institute of Technology, Nara College

表 1 TaskPit により計測されるデータの例

タスク名	開始時刻	終了時刻	左クリック	右クリック	打鍵回数	実行ファイル名
TaskPit	2013年05月10日 15時21分54秒	2013年05月10日 15時22分08秒	1	0	0	taskpit.exe: TaskPit
ブラウザ	2013年05月10日 15時22分08秒	2013年05月10日 15時22分11秒	1	0	0	firefox.exe: #886 (track)にuploadした設計書の履歴情報について? Solution - Mozilla Firefox
ブラウザ	2013年05月10日 15時22分11秒	2013年05月10日 15時22分19秒	1	0	0	firefox.exe: VC知??-VC?答 - Mozilla Firefox
TaskPit	2013年05月10日 15時22分19秒	2013年05月10日 15時22分20秒	1	0	0	taskpit.exe: TaskPit
ファイル操作	2013年05月10日 15時22分20秒	2013年05月10日 15時22分49秒	1	0	0	explorer.exe: Program Manager
ファイル操作	2013年05月10日 15時22分49秒	2013年05月10日 15時22分51秒	2	0	0	explorer.exe: 解析系

### 3. 提案方法

次のステップを通して開発作業の推定を行う。

- ・ (ステップ 1) 実行ファイル名とウィンドウに含まれる文字列を、可能性のある開発作業に (1 対多関係のルールとして) ひも付ける。このルール作成は手作業により行う。
- ・ (ステップ 2) ステップ 1 で作成したひも付けのルールに基づいて、TaskPit の計測データ (表 1 に例示するもの) の各行に対して、可能性のある開発作業をラベル付けする。このステップ 2 は、ひも付けのルールに基づいて自動的に行う。
- ・ (ステップ 3) TaskPit の計測データの各行に対して、前後の開発作業、時間、打鍵数より、開発作業を一意に推定する。この推定は自動で行うが、今後、自動推定のためのルールを決める必要がある。

ステップ 1 では、図 2 のような対応表を作成する。図 2 の対応表では実行ファイルが excel.exe でウィンドウに「漢字コード」という文字列が含まれている場合、可能性のある開発作業として「テスト」と「デバッグ」を対応させている。このように可能性のある対応関係を列挙し、ルールとして記述する。

ステップ 2 では、ステップ 1 で作成したルールに基づいて、各時刻に対して可能性のある開発作業をラベルづけする。この時点では、複数のラベルが付与される場合があり、また、ルールに記載のないアプリケーションの実行についてはラベル付けが行われない。

ステップ 3 では、例えば、TaskPit の計測データの連続する 3 行のラベルが「プログラミング」「プログラミングまたはデバッグ」「プログラミング」となっていた場合、2 行目は前後の関係から「プログラミング」であると推定する。

提案方法は、従来の機械学習による方法[4]と比べて、手作業の労力が少なくて済むと期待される。従来法では機械学習のための学習データとして、数千行におよび TaskPit データのすべての行にラベル付けを手動で行う必要があるのに対し、提案方法は、表 2 のようなルールを記述すればよく、また一度記述したルールは異なるプロジェクトに対し

表 2 実行ファイル名・ウィンドウに含まれる文字と可能性のある開発作業の対応の例

実行ファイル	ウィンドウに含まれる文字列	可能性のある開発作業
excel.exe	漢字コード	テスト デバッグ
	報告書	事務
iexplore	不具合	テスト
devenv.exe	—	プログラミング デバッグ

でもある程度再利用できると期待される。

### 4. 今後の予定

現時点では、あるソフトウェア開発企業における 2 名、一週間分の開発作業のデータを用いてステップ 1 を行い、ひも付けルールを作成し、ステップ 2 を実施したところである。今後、ステップ 3 における推定のルールを決定し、手作業による推定と比較した精度評価を行う予定である。

### 5. 参考文献

- [1] T. DeMarco, "Controlling Software Projects: Management, Measurement & Estimation," Yourdon Press, New York, USA, 1982.
- [2] 独立行政法人情報処理推進機構ソフトウェア・エンジニアリング・センター, "IT プロジェクトの「見える化」中流工程編," 日経 BP, 2008.
- [3] W. S. Humphrey, "パーソナルソフトウェアプロセス入門," 共立出版, May 2001.
- [4] 大橋亮太, 上野秀剛, 門田暁人, 荒木健史, 山田欣吾, 松本健一, "自動計測データと機械学習に基づくソフトウェア開発の作業目的の推定," コンピュータソフトウェア, Vol.33, No.2, pp. 139-150, June 2016.
- [5] ソフトウェア開発行動記録システム TaskPit, <http://taskpit.jpg.org/>
- [6] P. Suthipornopas, P. Leelaprute, A. Monden, H. Uwano, Y. Kamei, N. Ubayashi, K. Araki, K. Yamada, K. Matsumoto, "Industry Application of Software Development Task Measurement System: TaskPit," IEICE Transactions on Information and Systems, Vol.E100-D, No.3, pp.462-472, Mar. 2017