



---

# 卒業研究報告書

平成23年度

---

研究題目

時系列に着目した  
プロセス改善のための作業目的分類

---

指導教員 上野秀剛 助教

---

氏名 大橋亮太

---

平成24年1月26日 提出

奈良工業高等専門学校 情報工学科

# 時系列に着目した

## プロセス改善のための作業目的分類

上野研究室 大橋 亮太

ソフトウェア開発者や開発チームが自身の開発能力向上や開発プロセスの改善をするための手法である Personal Software Process /Team Software Process (PSP/TSP) が Watts Humphrey らによって提唱されている。PSP/TSP は開発者および開発チームの作業履歴を記録し、コーディングやデバッグ、設計、会議といった個々の開発作業にどれだけの時間を費やしているかを計測し、効率改善すべき作業の発見や、開発プロセスの効率化を行う手法である。

そこで PSP/TSP での作業履歴計測を支援するためのツールである TaskPit がある。しかし TaskPit では、記録される作業に対して、その作業が何を目的として行われているのか(作業目的)分からず、計測された作業履歴が PSP/TSP に活かしにくいといった問題がある。

本研究の目的は、コンピュータ上で行われ、TaskPit で記録される作業が、複数ある作業目的のうち、どれに属するかを自動的に判別することである。記録される作業の目的が分かることで、どの作業目的に時間を費やしているのかが分かり、計測された作業履歴を PSP/TSP に適用しやすくなると考えられる。そのための手法として、コンピュータ上での作業履歴を計測するシステムを用い、計測された作業履歴から、時系列に着目した上で機械学習を用いて分類する手法を提案する。

提案手法では、まず、一定期間の作業履歴を計測する。次に、計測された作業履歴から、作業の順序関係の特徴として用いることができるように処理を施す。最後に、処理を施した作業履歴データを学習データとして機械学習に学習させ、作業履歴における作業を作業目的に分類する分類器を構築する。分類器とは、膨大なデータから有益な知識を抽出するデータマイニングにおいて、データを分類するアルゴリズムである。提案手法の有用性を確認するために行った実験は、5名の被験者に5日間の作業履歴を計測してもらい、その作業履歴を用いて提案手法を実際に適用して、時系列に着目した場合としない場合の分類結果を比較した。

実験の結果、提案手法は時系列に着目しない場合の分類結果よりも、高い精度で分類できることが分かった。

# 目次

<b>1</b>	<b>はじめに</b>	<b>1</b>
<b>2</b>	<b>関連研究</b>	<b>2</b>
2.1	PSP/TSP 計測支援システム	2
2.2	作業目的の分類	2
<b>3</b>	<b>提案手法</b>	<b>4</b>
3.1	作業の時系列	4
3.2	TaskPit	5
3.3	分類手法	7
<b>4</b>	<b>実験</b>	<b>8</b>
4.1	被験者と計測環境	8
4.2	計測する作業	8
4.3	作業目的	8
4.4	分類機による分類	10
<b>5</b>	<b>結果と考察</b>	<b>13</b>
5.1	全体の結果	13
5.2	異なる傾向を示した被験者	14
5.3	特徴の重要度	16
5.4	考察	17
<b>6</b>	<b>おわりに</b>	<b>19</b>
	<b>参考文献</b>	<b>21</b>

## 1 はじめに

ソフトウェア開発者や開発チームが自身の開発能力向上や開発プロセスの改善をするための手法である Personal Software Process / Team Software Process (PSP/TSP) が Watts Humphrey らによって提唱されている [1]。PSP/TSP は開発者および開発チームの作業履歴を記録し、コーディングやデバッグ、設計、会議といった個々の開発作業にどれだけの時間を費やしているかを計測し、効率改善すべき作業の発見や、開発プロセスの効率化を行う手法である。

初期に提案された PSP/TSP では、作業の内容が変化するとともに、作業時間をストップウォッチで計測し、紙などに記録するという方法がとられていた。そのため、計測を支援するシステムが複数提案されている [2][3][4]。しかし、これらのシステムは作業時間の計測と記録を自動的に行える一方で、遂行中の作業が変化した際に手動で入力する必要がある。そのため、作業を切り替えた際に入力を忘れてしまう、計測に気を取られ作業に集中できないといった問題がある。

そこで、作業の計測を自動化するためのツールである TaskPit[6]がある。TaskPit は、使用しているアプリケーション名とウィンドウ名から一つの作業を特定し、作業時間やクリック数、打鍵数と共に記録する。例えば、WINWORD.EXE がウィンドウ名“設計書.docx”で起動されているとき、設計をしているものとして記録する。

しかし、“設計書.docx”が開かれても、コーディングをしている場合と、設計をしている場合など、複数の目的が考えられる。すなわち、記録される作業と、その作業が何を目的として行われているかが1対1に対応していない。

このように、実際の作業においては、一つの作業が複数の作業目的に関連付けられる。プロセスの改善は、作業自体に費やした時間ではなく、作業目的に対して費やした時間を計測する必要がある。そのため、ある作業が複数ある作業目的のうちどれに分類されるのかを判別する必要がある。

そこで本論文では、開発者が行う作業の順序に着目し、分類器の一つである Random Forests を用いた分類手法を提案する。また、本校の情報工学科5年生を対象に実験を行い提案手法の有効性について考察、検討する。

## 2 関連研究

### 2.1 PSP/TSP 計測支援システム

Watts らによって提唱されている PSP/TSP は、開発者および開発チームが自分たちの作業に関するデータを収集し、分析することで、プロセスを改善する手法である。PSP/TSP では、各作業に必要な時間を事前に見積もり、計測したデータから見積もりとの差異、およびその原因を分析することでプロセスの問題点を明らかにする。

PSP/TSP が提唱された当初(1996年)は、作業履歴の計測は紙を用いた手作業によって行われており、計測の際に、多大なコストがかかるとされてきた。そのため、これまでに Process Dashboard[2]、Task Coach[3]、Slim Timer[4] などの計測支援システムが提案されている。しかし、これらのシステムは作業時間の計測と記録を自動的に行える一方で、遂行中の作業が変化した際に手動で入力する必要がある。そのため、作業を切り替えた際に入力を忘れてしまう、計測に気を取られ作業に集中できないといった問題があり、広く普及するに至っていない。

そこで、作業の計測を自動化するためのツールである TaskPit[6]がある。しかし、Task Pit では計測される作業とその作業に対する目的が対応しておらず、開発プロセスの改善に役立てにくい点が問題となっている。

本研究では、TaskPit で計測される作業の履歴から、分類器を用いて作業を目的に自動分類することで計測作業を容易にする。

### 2.2 作業目的の分類

Michael らはコンピュータ上での作業を作業目的に分類する際、どのような機械学習法を用いると高精度の分類ができるのか、また、どのような特徴を用いることで高精度の分類ができるのかについて述べている [5]。Michael らは、アプリケーション名やウィンドウタイトル、コンテンツ(閲覧中の Web ページ内のテキストデータ、編集集中である PowerPoint のテキストデータ)を特徴とし、Naive Bayes(NB)、Linear Support Vector Machine(SVM)、k-Nearest Neighbor(k-NN)を用いて、どの特徴とどの学習法の組み合わせで高い精度が得られるかを実験により確認した。Michael らの報告によると、ウィンドウタイトルとコンテンツを特徴として、NB、k-NNを用い分類を行うと精度の高い結果が得られた。

Michael らの研究ではコンテンツという要素を特徴としており、プライバシーに配慮していない点がある。コンピュータ上では私的な作業も行われるため、閲覧中の Web ページ内容などを記録すると、計測者に対するプライバシーの侵害になると考えられる。さらに、全てのアプリケーションの利用履歴を記録しており、実際のシステムで用いる際、計測者に「監視されている」という感覚を与え、システム導入の妨げになる恐れがある。

また、特徴はその作業に関するもののみを使用しており、どのような特徴の組合せを用いても、複数の作業目的に対する作業の特徴と被る可能性が考えられるため、分類精度に限界があると考えられる。例えば、仕様書を開いて作業している場合でも、ソフトウェアの開発工程である設計を作業目的とする場合と、実装を作業目的とする場合があり、アプリケーション名やウィンドウタイトルが同じため、判別ができない可能性がある。

本研究では、計測者のプライバシーを侵害せず、時系列に着目した特徴を用いて分類を行うことで、より高い精度の分類を試みる。

### 3 提案手法

本章では、作業の時系列に着目して Random Forests を用いた作業目的の分類手法について述べる。以下、提案手法で用いる時系列の考え方、TaskPit について説明し、最後に分類手法について述べる。なお、本論文ではテキスト閲覧・編集やブラウジングなどを作業、このような作業が何のために行われているかを作業目的と呼ぶ。

#### 3.1 作業の時系列

一つの作業目的は、アプリケーション上での作業の集まり（作業群）であると考えられ、式 1 の形で表すことができる。

$$\text{作業目的} = [\text{作業 } A, \text{作業 } B, \text{作業 } C, \dots] \quad (1)$$

例えば、ソフトウェアの開発工程である設計、実装、テストでは以下のような作業群で構成されると考えられる。

設計 = [仕様書上での作業, 設計書上での作業]

実装 = [仕様書上での作業, 設計書上での作業, 開発環境上での作業]

テスト = [仕様書上での作業, 設計書上での作業, 開発環境上での作業]

これらの作業群は、作業目的ごとに特有のパターンがあると考えられる。そのため、このような作業群を抽出することで、作業目的と作業の対応関係を自動的に抽出できると考えられる。

作業目的を判別するために、作業順序に着目する。各作業目的には作業の順序にも特有のパターンが存在するはずであり、そこに着目することで、作業目的を判別できると考えられる。作業の順序関係に着目したとき、設計、実装、テストでは図 1 のような順序関係があると考えられる。図中の左側の文字列が、右側の各作業群の目的を表す。四角形は作業を表し、文字列は作業の作業名、マウス、キーボードのアイコンは作業内でのクリック、打鍵回数の多さを表す。矢印は作業の流れを示す。

設計では、実装、テストと比較して作業が一つ少なく、異なる作業内容、順序から設計に関する作業を分類することができる。実装とテストは作業群が同じであるが、開発環境上での作業で、クリック回数や打鍵回数に違いがあると想定され、この違いから作業をそれぞれに分類できると考えられる。また、人では気づくことのできない順序関係が隠れている可能性もある。

本研究では、上記の仮定に基づき、分類器を用いて時系列データから作業目的を分類することを試みる。

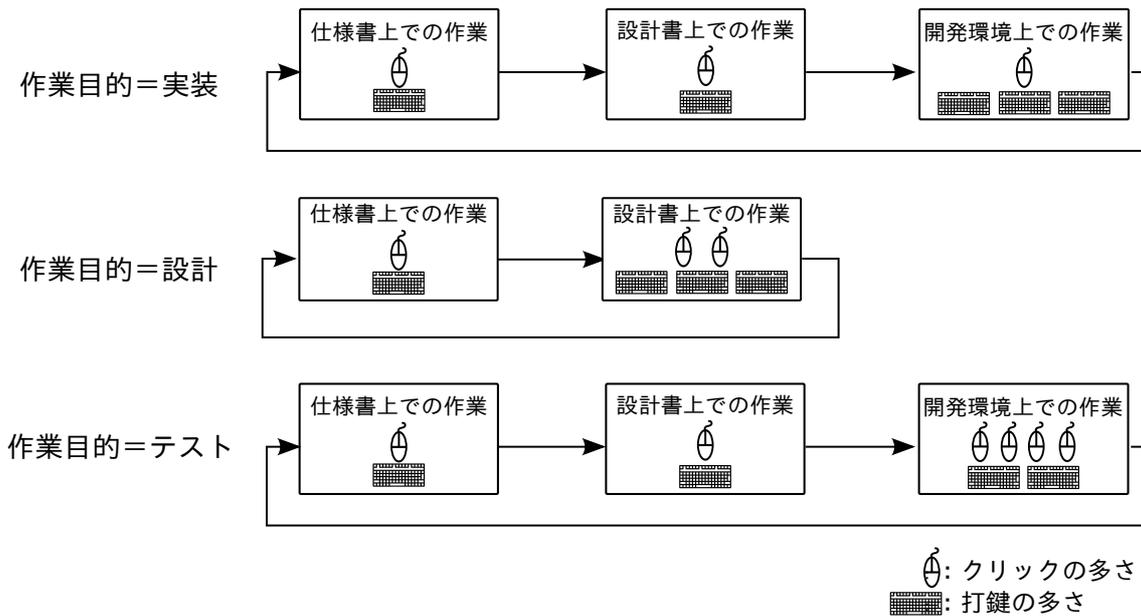


図 1 作業順序

### 3.2 TaskPit

TaskPit は門田らによって提案された PSP/TSP 支援計測支援システムである。図 2 に TaskPit のインタフェースを示す。

本システムでは、あるアプリケーションに対する操作を 1 つの作業とみなし、各アプリケーション上で行われた作業の時間を記録することで、各作業に費やした時間を計測している。これによって、例えば、Word を使った「テキスト閲覧・編集」、Eclipse を用いた「コーディング」といった粒度で作業を計測することができる。また、Word やブラウザなど、多数の目的で使用されるアプリケーションは、ウィンドウ名を見ることで、より細かい粒度で作業を計測することができる。例えば、Word を用いた作業で、“spec”のウィンドウ名が表示されていれば、作業名を「仕様書閲覧・編集」として計測することが可能である。このようなアプリケーションと作業とのひも付けは、TaskPit の設定ファイルで行うことができる。設定例を図 3 に示す。

また、TaskPit はチームとして計測を行う際、計測者に対してのプライバシーと個人情報保護に配慮しており、キーストロークの内容、アプリケーションの動作内容、ウィンドウ名、ファイル名、ファイルの内容等については記録しない。そして、全てのアプリケーションの使用履歴を記録することは、「監視されている」という感覚を計測者に与え、システム導入の妨げになる恐れがあるため、「登録外」として計測される。

TaskPit が計測するログデータの例を図 4 に示す。ログには、作業名、作業開始時間、作業終了時間、右クリック回数、左クリック回数、打鍵数が含まれる TaskLog

というファイルがある。本研究では、TaskPit で計測した作業履歴を用いた分類手法を提案する。



図 2 TaskPit

テキスト閲覧・編集 = TeraPad.exe|notepad.exe|WINWORD.EXE|Acrobat.exe|LabyTex.exe  
 プレゼン編集 = POWERPNT.EXE  
 データ分析 = EXCEL.EXE|Rgui.exe|javaw.exe:Weka|javaw.exe:GemX  
 コーディング = eclipse.exe|devenv.exe|EXCEL.EXE:Microsoft Visual Basic  
 デバッグ = devenv.exe:実行中  
 TaskPit = TaskPit.exe|TaskAnalyzer.exe  
 ブラウザ = iexplore.exe|firefox.exe|Safari.exe|chrome.exe  
 メール = thunderbird.exe|Outlook.exe|iexplore.exe:gmail

図 3 アプリケーションと作業のひも付け

作業目的, 開始時刻, 終了時刻, 左クリック, 右クリック, 打鍵回数  
 ファイル操作, 2011年12月20日13時25分35秒, 2011年12月20日13時25分48秒, 0, 0, 0  
 TaskPit, 2011年12月20日13時25分48秒, 2011年12月20日13時25分54秒, 3, 0, 0  
 ファイル操作, 2011年12月20日13時25分54秒, 2011年12月20日13時26分09秒, 2, 0, 0  
 デスクトップ, 2011年12月20日13時26分09秒, 2011年12月20日13時26分27秒, 1, 0, 0  
 ファイル操作, 2011年12月20日13時26分27秒, 2011年12月20日13時26分36秒, 5, 0, 2  
 データ分析, 2011年12月20日13時26分36秒, 2011年12月20日13時29分41秒, 24, 0, 64  
 デスクトップ, 2011年12月20日13時29分41秒, 2011年12月20日13時29分42秒, 2, 0, 0  
 TaskPit, 2011年12月20日13時29分42秒, 2011年12月20日13時29分43秒, 1, 0, 0  
 データ分析, 2011年12月20日13時29分43秒, 2011年12月20日13時29分45秒, 2, 0, 0

図 4 ログファイル

### 3.3 分類手法

本研究では他の分類器よりも高精度であるとされている Random Forests[7] を用いる。Random Forests は、木構造の条件分岐で予測を行う決定木を弱学習器として、作成した決定木の集団学習により予測を行い、分類の精度を向上させる機械学習アルゴリズムである。集団学習とは、精度の高くない分類器である弱学習器を複数組み合わせることで、高精度な分類器を構築する方法である。また機械学習とは、人間が自然に行っている学習と同様の機能をコンピュータで実現するための技術、手法である。そして、多種多様なデータの中から、規則性・パターン・知識を発見し、現状の把握や将来の予測に役立てるのが機械学習の目的となる。機械学習アルゴリズムは上記の目的を達成するために考案されたものである。本研究では、分類には統計処理用の言語である R 言語及びそのパッケージを用いる。Random Forests では組み合わせるベース分類器の数  $ntree$  と分岐の際に用いる説明変数の数  $mtry$  を調整することができる。本論文内で行われる実験では、説明変数の総数を  $M$  としたとき、Breiman の推奨する  $ntree = 500$ 、 $mtry = \sqrt{M}$  とする。Random Forests のアルゴリズムを次に示す。ここで与えられるデータセットは、 $P$  件の作業目的と、 $M$  件の作業特徴とする。

1.  $P$  件のオリジナルのデータセットから重複を許し、ランダムに  $P$  件を抽出し、学習用データセットとする。ここで作成された学習用データセットはブートストラップサンプルと呼ばれ、このデータセット作成法をブートストラップサンプリングと呼ぶ。この作業を規定の回数繰り返す。ここでは  $N$  回とする。与えられたデータセットから  $N$  組のブートストラップサンプル  $B_1, B_2, \dots, B_i, \dots, B_N$  を作成する。ブートストラップサンプルは変数をサンプリングして作成する。その際、用いるデータセットの約  $1/3$  はテスト用として取り除き、残りを学習用とする。テスト用として取り除いたデータを OOB (Out-Of-Bag) データと呼ぶ。
2. 各々のブートストラップサンプル  $B_i$  を用いて未剪定の最大の決定木  $T_i$  を生成する。通常、決定木では木が大きくなり成長するのを防ぐため、情報量の少ない不適切な枝を除去し、木を対象データに対して最適化する剪定を行う。しかし、剪定を行った決定木は弱学習器にならないため、Random Forests では剪定を行わない。その後、木の生成に用いていない OOB データを用いてテストを行う。その誤り率を OOB 推定値と呼ぶ。 $T_i$  の構築を行う際の各分岐ノードは、異なる木を多数生成するため、 $mtry$  個の変数をサンプリングし、その中から最も分岐が良い変数を用いる。
3. 分類器は、すべてのブートストラップサンプル  $B_i$  の OOB 推定値に基づいて多数決を取る。

## 4 実験

5名の被験者にコンピュータ上での作業をTaskPitを用いて計測してもらい、計測されたデータから提案手法を用いた分類の精度を評価する実験を行った。

### 4.1 被験者と計測環境

情報工学科5年生の5名に対して、所属研究室で利用しているコンピュータにTaskPitをインストールしてもらい、作業を計測する。作業は被験者らの研究室のコンピュータ上で行われる作業を全てTaskPitを用いて計測する。作業を計測する期間は12/16~12/22の5日間(土日除く)とした。土・日曜日を除いた理由は、この曜日に研究室内で作業が行われていなかったためである。コンピュータの環境OSとCPUは、5台のうち3台がWindows7、インテル Core2 Duo プロセッサ E7500、残りの2台がWindows Vista、インテル Core2 Duo プロセッサ E6850を搭載していた。

### 4.2 計測する作業

計測対象とする作業は、Microsoft OfficeのWordやテキストエディタソフトを用いた作業を「テキスト閲覧・編集」、Microsoft VisualStudioやEclipseといった統合開発環境での作業を「コーディング」というようにひも付けを行い、計測する。これは、事前に被験者に用意してもらったアプリケーションのリストと作業のリストから著者が設定する。事前に設定されなかったアプリケーションを用いて作業は「登録外」として記録する。

### 4.3 作業目的

本研究では分類器を作成するために、機械学習の教師あり学習を用いる。教師あり学習では、答えを持ったデータが必要になる。本研究で答えとなるのは作業に対する作業目的である。そのため、現在のTaskPitに手動で作業の目的を切り替え、記録する機能を図5のように実装した。この機能によって、研究に用いるログファイルの項目に作業目的が追加され、図6のように記録される。

作業目的には実験期間中に起こり得る目的を著者が用意し、目的は、研究(卒論・予稿)、研究(発表)、研究(コーディング)、研究(調査)、研究(その他)、課題(信号処理)、課題(ソフトウェア工学)、課題(その他)、息抜き、その他、を設定する。また、TaskPit起動直後、作業目的が設定されていない状況(入力忘れ)を想定して、デフォルトという項目を追加した。これらの作業目的の設定根拠であるが、まず、研究室で行われる作業の作業目的として、「研究」「課題」、「息抜き」とこれらに属さない「その他」があると考えられる。ここで「研究」と

「課題」についてはPSP/TSPを用いる際に、研究の何に、どのような課題に時間をかけているのかが分からなければ、作業履歴をプロセス改善に役立てることが難しく、それらについて知る必要があるため、細分化を行う。研究についての作業目的は、計測時期を考慮すると、卒論・予稿の作成、実験ツール等作成のためのコーディング、実験手法やその他の研究に必要な知識を得るための調査、研究室での輪行資料作成のための発表等が想定される。課題についての作業目的は、計測時期に課題として出ていたものが対象となると想定される。以上のような考えから作業目的の設定を行った。

実験で得られた作業履歴の例を図6に示す。提案手法の評価は、手法により予測された作業目的が、被験者が入力した作業目的と合致しているかを見る。評価方法の詳細は4.4節で説明する。

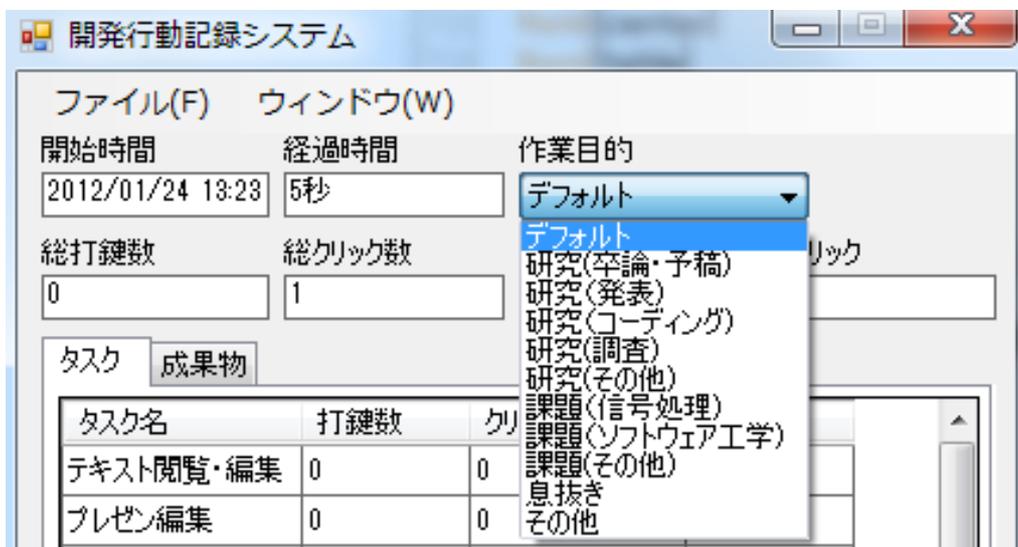


図5 改良した TaskPit

作業, 左クリック, 右クリック, 打鍵回数, 作業目的  
 TaskPit, 19, 12, 0, デフォルト  
 TaskPit, 3, 0, 0, 課題(ソフトウェア工学)  
 ブラウザ, 35, 3, 153, 課題(ソフトウェア工学)  
 登録外, 2, 0, 0, 課題(ソフトウェア工学)  
 ブラウザ, 12, 0, 0, 課題(ソフトウェア工学)  
 ファイル操作, 9, 0, 0, 課題(ソフトウェア工学)  
 テキスト閲覧・編集, 9, 0, 4, 課題(ソフトウェア工学)  
 ファイル操作, 1, 0, 0, 課題(ソフトウェア工学)  
 テキスト閲覧・編集, 1, 0, 0, 課題(ソフトウェア工学)

図6 作業目的の追加/計測履歴の例

#### 4.4 分類機による分類

実験では、作業順序に着目する場合と、着目しない場合とを比較し、分類精度にどのような差が表れるか、また分類時に前後いくつの作業を用いるかを変化させ、どのような差が出るかを明らかにする。そのために、前処理後のデータと、それを加工したデータセットの3種、合わせて4種類のデータセットを用意する。4種類のデータセットの内訳は、1) 前処理後のログファイル、2) 前処理後のログファイルに、前後1つ迄の作業名、右クリック回数、左クリック回数、打鍵数を加えたもの、3) 前処理後のログファイルに、前後2つ迄の作業名、右クリック回数、左クリック回数、打鍵数を加えたもの、3) 前処理後のログファイルに、前後3つ迄の作業名、右クリック回数、左クリック回数、打鍵数を加えたもの、となる。分類実験は、データセットの半分を学習用データ、残りの半分をテスト用データとする 2-fold-cross-validation を用いて行う。分類結果は表1のようにして出力される。表1において、列の項目は被験者らが設定した正しい作業目的を表しており、行の項目は分類器による分類先を表している。列と行の項目が一致しているセルは、分類器による分類が正しいことを示す。列と行の項目が一致していないセルは、分類器が誤った分類をしていることを示す。それぞれのセル内の数字は、その列の項目を作業目的とする作業を行の項目に分類した数を示す。

分類精度の評価には、recall(再現率)、precision(精度)、F1-value(F1 値)の三種の評価尺度を用いる。ある作業目的  $X$  とそれ以外の作業目的における分類の結果は表2で表すことができる。表の値は以下のような意味を表す。

- $A$ : 被験者が設定した作業目的  $X$  を持つ作業に対し、分類器が正しく作業目的  $X$  に分類できた数
- $B$ : 被験者が設定した作業目的  $X$  以外を持つ作業に対し、分類器が誤って作業目的  $X$  に分類した数
- $C$ : 被験者が設定した作業目的  $X$  を持つ作業に対し、分類器が誤って作業目的  $X$  以外に分類した数
- $D$ : 被験者が設定した作業目的  $X$  以外を持つ作業に対し、分類器が作業目的  $X$  以外に分類した数

表1 分類結果の出力例

		設定された作業目的			
		課題(ソフトウェア工学)	課題(信号処理)	息抜き	研究(卒論・予稿)
予測結果	課題(ソフトウェア工学)	167	0	41	33
	課題(信号処理)	0	14	1	0
	息抜き	56	7	282	54
	研究(卒論・予稿)	7	0	35	140

表 2 評価値の算出法

		設定された作業目的	
		作業目的 X	作業目的 X 以外
予測結果	X	A	B
	X 以外	C	D

表 2 を用いて，評価に用いる尺度と計算方法について説明する．

分類器が分類した作業目的  $X$  を目的とする作業のうち，何割が実際に作業目的  $X$  を目的とした作業であるかを表す precision は， $A, B$  より，式 1 によって求められる．

$$precision = \frac{A}{A + B} \quad (2)$$

precision は  $0 \sim 1$  の範囲をとり，値が高ければ高いほど分類器が実際の作業目的  $X$  に対して，別の作業目的に分類していないことを示す．

被験者が設定した作業目的  $X$  を目的とする作業のうち，何割を分類器が作業目的  $X$  を目的として分類できているかを表す値 recall は， $A, C$  より，式 3 によって求められる．

$$recall = \frac{A}{A + C} \quad (3)$$

recall は  $0 \sim 1$  の範囲を取り，値が高ければ高いほど分類器が，被験者が設定した作業目的  $X$  を目的とする作業を，作業目的  $X$  として正しく分類できていることを示す．

recall とは取りこぼしの少なさを表す指標であり，precision とは分類した際の間違いの少なさを表す指標である．ある作業目的  $X$  に分類した際の間違いを少なくしようとするとき，確実に作業目的  $X$  である作業だけを作業目的  $X$  に分類すればよい．このとき，曖昧なものは全て分類されないため，取りこぼしが増えると考えられる．同じように，ある作業目的  $X$  に対して取りこぼしを少なくしようとするとき，作業目的  $X$  であると考えられる作業は全て作業目的  $X$  に分類すればよい．このとき，曖昧なものも分類されてしまうため，分類した際の間違いが多くなると考えられる．このように片方の値を向上させようとするとき，片方の値が小さくなってしまうため，これらの指標はトレードオフの関係性にあると言える．このため，分類器の分類精度について評価する指標として本研究では F1-value を用いる．F1-value は precision と recall の調和平均をとった値で，式 4 によって求められる．

$$F1 - value = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} \quad (4)$$

F1-value は  $0 \sim 1$  の範囲をとり，precision，recall が両者ともに高い場合に高い値を示す．また，F1-value は高ければ高いほど良い．

多くの機械学習法では乱数を使用しているため、同一のデータセットに対して分類を繰り返しても分類の精度が同じになるとは限らない。そこで、評価には実験を10回繰り返したときの、評価値の平均を用いる。

## 5 結果と考察

ここでは4章4節で述べた評価方法に基づいて、各被験者から得られたデータセットを分類に用いた際の結果を評価する。

### 5.1 全体の結果

表3に提案手法による分類結果を示す。実験では、被験者毎に計測された作業目的が異なるため、作業目的ごとに評価値を求め、それらの平均値を各被験者のデータセットに対する評価指標とした。表3の値は、各被験者のデータセットに対する評価指標の平均値をデータセットの種類ごとに求めたものである。また、各データセットの値を図7～図9に示す。

図7～図9から、前後の作業を多く見るほどすべての評価尺度で高い値を示すことがわかる。このことから、作業順序を特徴としてみることで分類精度が向上するといえる。特に、計測期間中に各被験者が頻繁に行っていた作業目的については高い精度で分類することができた。ただし、順序関係をなしから3つにむかって多く見るにつれ、評価尺度の上昇値は漸減している。このことは、分類器に渡す前後の作業を増やすほど、値が改善されるわけではないことを示唆している。

表3 分類結果

	前後なし	前後1つ	前後2つ	前後3つ
precision	0.521	0.693	0.752	0.778
recall	0.466	0.629	0.657	0.671
F1-value	0.492	0.659	0.701	0.721

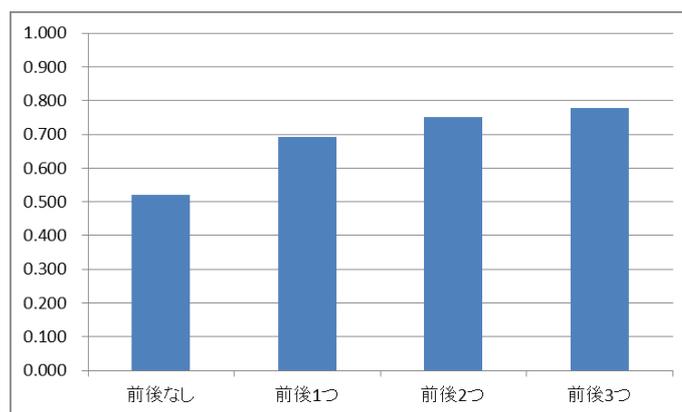


図7 全体の結果 precision

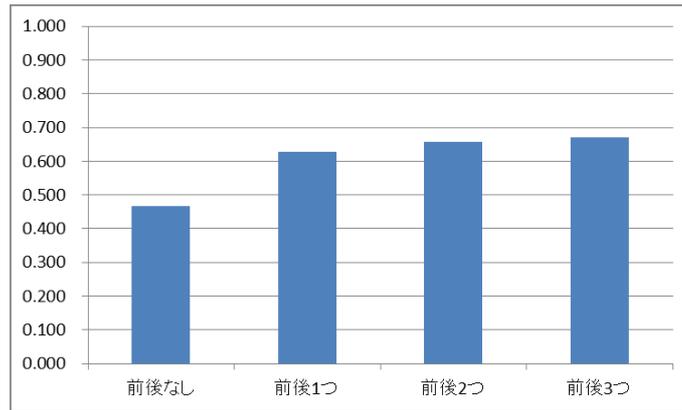


図 8 全体の結果 recall

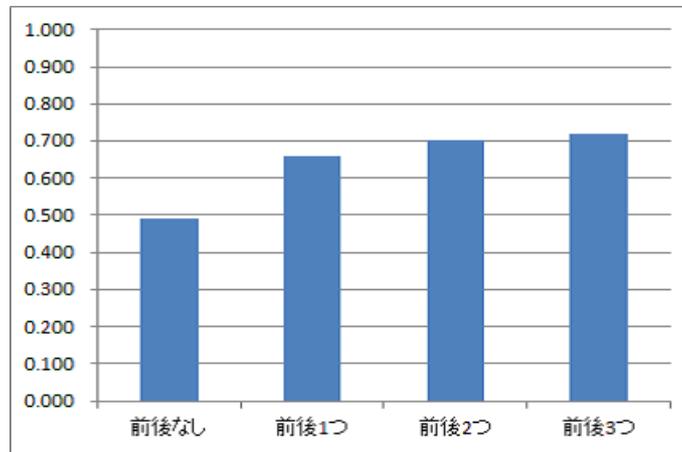


図 9 全体の結果 F1-value

## 5.2 異なる傾向を示した被験者

5人の被験者のうち、被験者3について他の4人と異なる傾向が見られた。被験者3の各作業目的に対する評価値を表4～表6に示す。

まず、表6から課題(ソフトウェア工学)のF1-valueが低く、全体の平均値を下げていることが分かる。課題(ソフトウェア工学)についてはrecallの値が他の作業目的よりも低い(表5)。個別の分類結果を調査すると課題(ソフトウェア工学)を目的とした作業の66.4%が、息抜きに分類されていた。

息抜きに分類されてしまう原因は、息抜きと課題(ソフトウェア工学)という作業目的に対して、「Twitter」と「ブラウザ」という作業を頻繁に、かつ交互に行っていることが挙げられる。その一例として図10と図11に息抜きと課題(ソフトウェア工学)を作業目的とした作業の履歴の一部示す。図10からは、作業の列を見ると「Twitter」と「ブラウザ」という作業が交互に、かつ頻繁に記録されていることが分かる。息抜きを目的とした作業からはこの傾向が多く見られた。図

11からも「Twitter」と「ブラウザ」という作業が交互に記録されている箇所がいくつかあることが分かる。

以上の点から提案手法では、異なる作業目的でも作業順序に似通っている部分がある場合、分類精度が下がる可能性がある。

また、評価実験後に被験者3にインタビューを行ったところ、課題(ソフトウェア工学)を作業目的とした「Twitter」は、実際は息抜きを作業目的とした作業であったことが分かった。このことから今回の作業履歴計測では、一部の作業に対して本来の作業目的とは異なる作業目的で計測されていることが分かる。

表4 被験者3 precision

	課題(ソフトウェア工学)	息抜き	研究(卒論・予稿)	平均
前後なし	0.399	0.529	0.554	0.494
前後1つ	0.537	0.620	0.763	0.640
前後2つ	0.593	0.637	0.817	0.682
前後3つ	0.681	0.621	0.841	0.714

表5 被験者3 recall

	課題(ソフトウェア工学)	息抜き	研究(卒論・予稿)	平均
前後なし	0.119	0.642	0.670	0.477
前後1つ	0.322	0.794	0.768	0.628
前後2つ	0.313	0.862	0.786	0.654
前後3つ	0.259	0.901	0.777	0.646

作業, 左クリック, 右クリック, 打鍵回数, 作業目的

Twitter,1,0,0, 息抜き  
 ブラウザ,1,0,1, 息抜き  
 Twitter,3,0,7, 息抜き  
 ブラウザ,11,0,14, 息抜き  
 Twitter,1,0,8, 息抜き  
 ブラウザ,26,0,0, 息抜き  
 Twitter,1,0,0, 息抜き  
 ブラウザ,53,10,52, 息抜き  
 Twitter,1,0,0, 息抜き  
 ブラウザ,1,0,0, 息抜き  
 Twitter,1,0,0, 息抜き  
 ブラウザ,1,0,0, 息抜き  
 Twitter,1,0,0, 息抜き  
 ブラウザ,19,2,1, 息抜き  
 Twitter,1,0,5, 息抜き  
 ブラウザ,1,0,0, 息抜き  
 Twitter,2,0,74, 息抜き

図10 息抜きの作業順序(一部)

表 6 被験者 3 F1-value

	課題 (ソフトウェア工学)	息抜き	研究 (卒論・予稿)	平均
前後なし	0.184	0.580	0.606	0.486
前後1つ	0.403	0.697	0.766	0.634
前後2つ	0.409	0.733	0.801	0.668
前後3つ	0.376	0.735	0.808	0.678

作業, 左クリック, 右クリック, 打鍵回数, 作業目的  
 Twitter, 1, 0, 0, 課題 (ソフトウェア工学)  
 ファイル操作, 1, 0, 0, 課題 (ソフトウェア工学)  
 ブラウザ, 3, 0, 0, 課題 (ソフトウェア工学)  
 Twitter, 1, 0, 2, 課題 (ソフトウェア工学)  
 ブラウザ, 1, 0, 0, 課題 (ソフトウェア工学)  
 ファイル操作, 4, 0, 0, 課題 (ソフトウェア工学)  
 コーディング, 1, 0, 0, 課題 (ソフトウェア工学)  
 テキスト閲覧・編集 or コーディング, 1, 0, 0, 課題 (ソフトウェア工学)  
 Twitter, 1, 0, 0, 課題 (ソフトウェア工学)  
 コーディング, 1, 0, 0, 課題 (ソフトウェア工学)  
 テキスト閲覧・編集 or コーディング, 1, 0, 0, 課題 (ソフトウェア工学)  
 ファイル操作, 2, 0, 0, 課題 (ソフトウェア工学)  
 コーディング, 1, 0, 0, 課題 (ソフトウェア工学)  
 テキスト閲覧・編集 or コーディング, 1, 0, 0, 課題 (ソフトウェア工学)  
 ブラウザ, 11, 0, 5, 課題 (ソフトウェア工学)  
 ファイル操作, 2, 0, 0, 課題 (ソフトウェア工学)  
 登録外, 0, 0, 0, 課題 (ソフトウェア工学)  
 ファイル操作, 0, 0, 0, 課題 (ソフトウェア工学)

図 11 課題 (ソフトウェア工学) の作業順序 (一部)

### 5.3 特徴の重要度

Random Forests では, 分類を行う際にどの特徴 (説明変数) が重要であったかをジニ係数を用いて寄与率として表すことができる。本実験では, 各被験者のデータセット毎に寄与率を算出した。寄与率の数値には各被験者毎に異なったが, どの特徴が重要であったかについては一定の規則があった。例として, 1人の被験者のデータセット (前後3つをみたもの) での各特徴の重要度を図12に示す。

提案手法で分類を行う際に一番重要とされていたものは, 各作業の作業名であった (例: テキスト閲覧・編集, データ分析)。次に重要とされていたのが左クリック数, 続いて打鍵数, 右クリック数であった。

よって, どのデータセットにおいても寄与率が低い右クリック数と, データセットによっては寄与率が低い打鍵数, 左クリック数を特徴から排除することで, 分類の精度が向上することも考えられる。

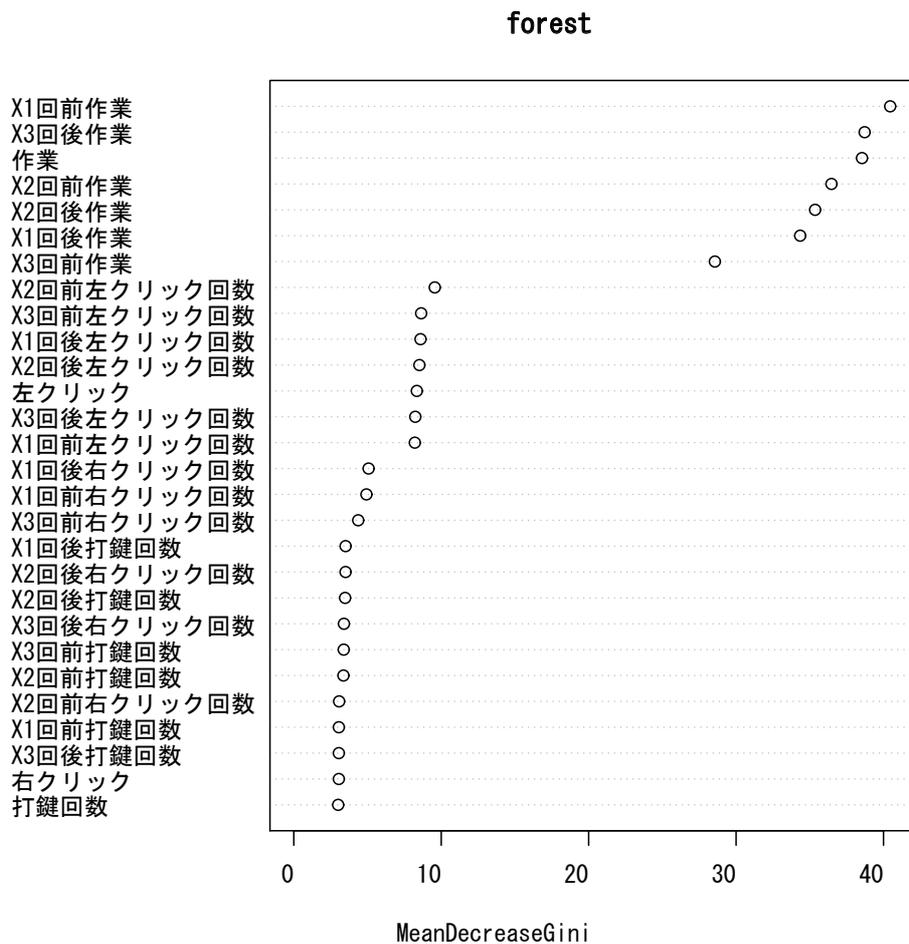


図 12 変数の重要度

#### 5.4 考察

実験の結果から、作業の順序関係を考慮することで、順序関係を考慮していないデータセットよりも、高い分類精度を得ることがわかった。さらに、計測期間中の主だった作業目的に関しては、特に高い分類精度が見られた。PSP/TSPでは多く時間をかけている作業目的を特定し、優先的に改善を行うことが重要であるため、特に有用であると考えられる。ただし、異なる作業目的同士で使用するアプリケーションや作業順序が似通っている場合は分類の精度が低くなることがあった。

この問題については、TaskPit で計測しているものの、提案手法では用いなかった要素（特徴）を用いることで改善できる可能性がある。例えば、TaskPit にはディレクトリ内のファイル増減を監視し、作業量を測定する機能がある。この作業量を用いることで、例えば設計書ファイルの容量が変化していれば、この作業

は設計を目的とした作業であるというように、他の作業目的との区別が明確につく特徴となり、分類精度の向上が期待できる。

また増やすのではなく、提案手法で用いた特徴を取り除くことで、例えば、重要度の低かった右クリック数を分類に用いないことで、分類精度が向上することも考えられる。

次に作業履歴計測の際に、一部の作業に対して本来の作業目的とは異なる作業目的で計測されてしまう問題が分かった。この問題に対する改善方法であるが、現在実装している機能では、作業目的が切り替わる毎に計測者の手によって正確に切り替えを行なってもらう以上の方法がない。よって、この問題点の解決方法の考案が必要となる。

## 6 おわりに

本研究ではTaskPitで計測される作業をその作業に対する目的に分類することを研究目的とし、目的到達のため、TaskPitが保存する作業履歴を用いて、時系列に着目し、機械学習法を用いて分類する手法を提案した。

提案手法では、改良したTaskPitを用いて計測した作業履歴と作業目的から、機械学習に学習させ分類器を作成する。

提案手法の有用性を確認する実験では、一週間の中に記録された作業目的のうち、主だったものに対する分類精度が高く、時間をかけている作業の目的を特定し、プロセス改善に役立てるPSP/TSPに対し、有用であると考えられる。

今後の展望としては、分類精度の向上を目指し、TaskPitの他の機能を利用した分類手法の発案や、どのような特徴の組み合わせを用いることで分類の精度が向上するか、より多くの被験者の作業履歴に対し分類実験を行い、データを取得すること、正確な作業履歴の計測手法の考案、実際にソフトウェア開発を行っている現場での作業履歴からの分類を行い、プロセス改善に役立てることができるか検証を行うことが挙げられる。

## 謝辞

本論文の執筆をおよび研究をすすめるに当たり、多くの方々に協力して頂きました。この場を借りてお礼を申し上げます。ありがとうございました。

指導教員である上野秀剛助教には、この1年間を通して研究に関する知識のご教授から、アドバイス、論文のチェックまで多くの面でご指導頂きました。また、私的な事柄についてもご相談を受けて下さり、アドバイスをいただくなど、一年間支えて頂きました。ここに深謝の意を表明させていただきます。ありがとうございました。

査読教員である山口賢一准教授からは、査読コメント、卒研発表会の両方で、鋭い質問、今後の研究に対するアドバイスをいただきました。ここに深謝の意を表明させていただきます。ありがとうございました。

同じ上野研究室であった皆様には、私の研究に対するアドバイスや発表での至らぬ点の指摘など、多くの面で支えて頂きました。ここに深謝の意を表明させていただきます。ありがとうございました。

同級生の皆様には、研究や課題などに追われる中、被験者として研究実験に協力して頂きました。ここに深謝の意を表明させていただきます。ありがとうございました。

## 参考文献

- [1] Humphrey , Watts S : “ パーソナルソフトウェアプロセス入門 , ” 共立出版 , May 2001 .
- [2] Process Dashboard , <http://processdash.sourceforge.net/pspdash.html>
- [3] Task Coach Your friendly task manager , <http://members.chello.nl/f.niessink/>
- [4] SlimTimer Time Tracking without the Timesheet , <http://www.slimtimer.com/>
- [5] Granitzer , Michael and Rath , Andreas S . and Kroll , Mark and Seifert , Christin and Ipsmiller , Doris and Devaurs , Didier and Weber , Nicolas and Lindstaedt , Stefanie N , “ Machine Learning based Work Task Classification ” , Journal of Digital Information Management , October 2009 pp.306-313(2009) .
- [6] 門田 暁人 , 亀井 靖高 , 上野 秀剛 , 松本 健一 , “ プロセス改善のためのソフトウェア開発タスク計測システム ” ソフトウェア工学の基礎 XV , 日本ソフトウェア科学会 FOSE2008 , pp.123-128, Nov. 2008.
- [7] Breiman, L , Random Forests, Machine Learning , Vol.45 , No.1 , pp.5 ~ 32 (2001) .