

開発者の所属企業規模を考慮したソフトウェア工学の有用性評価

村上 優佳紗
近畿大学大学院総合理工学研究科
m.yukasa@gmail.com

角田 雅照
近畿大学大学院総合理工学研究科
tsunoda@info.kindai.ac.jp

要旨

ソフトウェア工学の研究目標を設定する際、開発者の総数が多いボリュームゾーンのニーズを考慮する必要がある。日本では300人未満の中小企業及びそこに所属する開発者が多数を占めており、ここがボリュームゾーンといえる。ただし、ソフトウェア工学の最先端研究が、これらの企業に所属する開発者のニーズに込められているかどうかは明らかではない。そこで本研究では、ソフトウェア工学の最先端研究が、中小企業に所属する企業のニーズに対応できているかどうかを分析した。具体的には、ソフトウェア開発企業に所属する開発者に対し、最先端研究が自分の業務に有用であるかどうかを評価してもらった。その後、開発者が所属する企業の規模によりデータを層別して分析を行った。その結果、所属する企業の規模によって、業務に関連する研究カテゴリに差異が見られたが、研究の有用性及び興味の評価については、規模による差異は見られなかった。

1. はじめに

近年、ソフトウェアは企業の基幹業務システムや機器に組み込まれて広く利用されており、社会基盤として欠かせないものとなっている。このため、ソフトウェアに高い信頼性が求められるようになってきているが、その一方で、大規模なソフトウェアを短期間で開発することも同時に求められている。このようなソフトウェア開発が成功する確率を高めるために、ソフトウェア工学による開発の支援が求められる。

ソフトウェア工学による開発の支援を検討する際、開発者の総数が多いボリュームゾーンのニーズを考慮する必要がある。中小企業関連立法では、ソフトウェア業・情報処理サービス業について、資本金3億円以下または従業員300人以下を中小企業としている[2]。従業員300人未満の企業の割合は99%、それら

の企業においてソフトウェア開発に従事している者の割合は69%であり[7]、300人未満の中小企業及びそこに所属する開発者が、日本のソフトウェアベンチャーのボリュームゾーンであるといえる。

ただし、ソフトウェア工学の最先端研究が、ボリュームゾーンである、これらの企業に所属する開発者のニーズに込められているかは明らかでない。ソフトウェア工学の研究の有用性や方向性について検討した研究がいくつか存在するが[8][11]、これらの企業はFacebookやMicrosoftなど、規模の大きなソフトウェア開発会社に所属する開発者を対象としている。中小企業と大企業では、各ソフトウェアプロジェクトに従事するメンバー数や、開発対象のソフトウェアの規模が異なることが多いと考えられる。このため、所属する企業の規模が異なれば、技術者のニーズも異なる可能性がある。

本研究では、ソフトウェア工学の最先端研究が、中小企業に所属する企業のニーズに対応できているかどうかを分析する。ソフトウェア工学の最先端研究が大企業のニーズに込めることも重要であるが、ボリュームゾーンである中小企業のニーズの考慮も怠るべきでないと考えられる。分析では、ソフトウェア開発企業に所属する開発者に対し、ソフトウェア工学の最先端研究が自分の業務に有用であるかどうかを評価してもらった。その後、開発者が所属する企業の規模によりデータを層別して分析を行った。また、開発者による研究の有用性評価を支援するために、開発者の業務に関連する研究カテゴリとそれら属する研究の評価を、協調フィルタリングを用いて予測し、精度を評価した。

2. 関連研究

ソフトウェア工学の研究がソフトウェア開発に対して与えた影響について分析した取り組み (Impact Project) [9]では、プログラミング言語や構成管理ツ-

表 1 回答者の概要

従業員数	回答者数(人)	経験年数3年未満(人)	平均経験年数(年)	修士号取得者(人)
300人未満	6	4	6.7	2
300人以上1000人未満	5	5	1.2	2
1000人以上	5	5	1.2	1

表 2 回答者が関連する開発工程

従業員数	要求分析	基本設計	詳細設計	テスト	プログラミング	開発支援	プロジェクト管理
300人未満	33%	67%	67%	100%	83%	33%	17%
300人未満(経験年数3年以上除く)	0%	50%	50%	100%	75%	0%	0%
300人以上	20%	60%	60%	80%	80%	20%	0%
1000人未満	20%	60%	60%	100%	80%	0%	20%
1000人以上	20%	60%	60%	100%	80%	0%	20%

ルなどに対して、ソフトウェア工学がそれらの発展にどのように寄与してきたかを分析している[3][12]。その結果、この取り組みでは、研究が実際の開発方法に影響を与えていることを結論づけている。ここから、これまでのソフトウェア工学の研究に有用性があることはわかるが、最先端の研究も同様に有用なのか、また、どの程度の割合の研究が有用なのかは明らかではない。

ソフトウェア工学の研究の方向性について示唆した研究として、Rubinらのもの[11]が挙げられる。GoogleやFacebookなどの先端のソフトウェア開発会社に所属する開発者にインタビューし、彼らが直面する問題を明らかにするとともに、その問題に基づき、開発者が必要な情報に素早くアクセスすることをサポートする研究が望まれることなどを示している。ただし、この研究では、先端のソフトウェア開発企業以外の開発者のニーズは明らかにしていない。

本研究と類似した取り組みとして、2つの従来研究[1][8]が挙げられる。これらは同じ研究グループによって実施されたものであり、ソフトウェア開発企業に所属する開発者が、最先端の国際会議の論文の要約(1-2行)を読み、それに対して研究の価値があるかどうかを5段階で評価している。その結果、7割前後の研究が、価値があるという結果となっている。調査方法は基本的にZimmermannら[1][8]と同様の方法であるが、本研究とこれらの研究との違いは3つある。1つ目は各研究に対する評価の観点である。本研究では、「自分の業務に有用であるか」という観点で評価してもらっていることであり、これが従来研究との最も大きな違いである。2つ目は、所属する企業の規模の違いに着目して分析していることである(従来研究ではマイクロソフトなど巨大企業に所属する開発者が回答者の主体となっている)。3つ目は、個別の研究内

容に対して、より詳細な説明資料を読んでもらった上で評価してもらったことである。その結果、後述するように、上記従来研究と異なり、最先端の研究に対する有用性の評価は低くなっていた。

3. データ収集

分析で用いるデータは、ソフトウェア開発企業に所属する開発者から、アンケートフォーム(Googleフォーム)を用いて収集した。研究に対する有用性などの評価について、以下の手順に従って回答してもらった。

1. 業務に関する研究カテゴリを最大5つ(3つは必須)選ぶ。
2. 上位3件の研究カテゴリに関する論文の資料を読む。
3. 「業務に役立つか」、「興味を持ったか」について、それぞれ5段階(1が最低、5が最高)で評価する。

研究カテゴリの分類や、各カテゴリに含まれる研究論文については、ソフトウェア工学の国際会議で最も権威のあるICSE(International Conference on Software Engineering)の、2016年開催分のを参考にした。具体的には、会議の各セッションを研究カテゴリとした。研究カテゴリの詳細については、4.2節で後述する。各カテゴリ(セッション)には論文が4本程度含まれる。回答者の負担を減らすため、それらを直接読んでもらうのではなく、日本語の要約資料[6]を回答者に読んでもらった。資料は一本の論文につき2、3ページのスライド形式となっており、比較的読みやすい内容となっている。

また、各開発者について、以下の項目について回答してもらった。

表 3 研究カテゴリの概要及び回答者の業務に関連する割合

No.	カテゴリ名	論文数	総計	300人未満	300人未満 3年未満	300- 1,000人	1,000人 以上
1	Android (android で動くソフトウェアに関する研究)	4	2%				8%
2	Performance (ソフトウェアの性能)	4	4%			7%	8%
3	Empirical (ソフトウェア開発データの分析)	3					
4	Symbolic Execution (シンボリック実行によるソフト解析)	4					
5	Compilers and Emerging Trends (コンパイラ・最新トピック)	4					
6	Energy and Videos (ソフトウェア省電力化・ビデオの活用)	4					
7	Open Source (オープンソースソフトウェア)	2	4%	11%	8%		
8	Defect Prediction (ソフトウェアの欠陥予測)	4					
9	Synthesis (自動プログラム生成)	4	4%			7%	8%
10	API (Application Programming Interface)	3	4%	11%	8%		
11	Code Smells (コーディングの方法論)	4	2%				8%
12	Architecture (ソフトウェアの構造)	3	4%	6%		7%	
13	Testing (ソフトウェアのテスト)	8	20%	17%	17%	21%	23%
15	Effort Estimation and Search (開発工数見積もり)	3	2%	6%	8%		
16	Product Lines (プロダクトラインによるソフト再利用)	4					
17	Repair and Model Synthesis (プログラムの自動修正)	4					
18	Languages (ソフトウェア開発言語)	4	4%	6%	8%	7%	
19	Debugging (ソフトウェアのデバッグ)	4	9%	11%	8%	7%	8%
20	Requirements (ソフトウェアに対する要求)	4	7%	6%	8%	7%	8%
21	Dynamic Analysis (ソフトウェアの動的な解析)	4	2%			7%	
22	Security (セキュリティ)	4	2%	6%			
23	Collaborative (ソフトウェア開発の人的要因)	4	4%			7%	8%
24	Software Quality (ソフトウェアの品質)	3	9%	6%	8%	7%	15%
25	Program Analysis (プログラムの解析)	4					
26	Concurrency (ソフトウェアの並行実行)	4					
27	Maintenance (ソフトウェアの保守)	4	13%	17%	25%	14%	8%

- 取得学位 (学士, 修士など)
- 実務経験年数
- 所属する企業の従業員数 (300人未満, 300人以上, 1,000人未満, 1,000人以上)
- 担当する開発工程

4. 分析

分析の目的を明確にするために, 以下の4つのリサーチクエスチョンを設定した。

- RQ1: 業務に関連する研究カテゴリに偏りはあるか?
- RQ2: 所属する企業の規模によって, 関連する研究カテゴリに差異は生じるか?
- RQ3: 最先端研究の内容は業務に有用か?
- RQ4: 所属する企業の規模によって, 研究の有用性に差異は生じるか?

4.1. 回答者の概要

アンケートの回答者はソフトウェア開発企業に勤

務する16人である。回答者の概要について, 所属企業の従業員数別に層別したものを表1に示す。従業員数300人未満の会社の回答者グループでは, 開発経験年数が3年を超える回答者が2名いたが, その他のグループでは経験年数が3年未満であった。このため, 以降の分析において企業の規模別に回答傾向などを確かめる場合, 経験年数が3年以上の回答者を除外した場合についても示す。

業務に関連する工程: 回答者の業務に関連する工程について, 所属企業の従業員数別に層別したものを表2に示す。テストとプログラミングに関してはほとんどの回答者が関連し, 設計に関しても半数以上の回答者が関連している。要求分析や開発支援などについては, 300人未満の従業員数の企業の場合, 経験年数が浅い場合(表の2行目)は関連していないが, それ以外の規模の企業の場合, 割合は低いに関連している場合があった。

4.2. 業務に関連する研究カテゴリ

ICSEでの研究カテゴリを表3に示す。会議での各

セッションを各研究カテゴリとした。カッコ内は、セッションに含まれる研究の内容を考慮して筆者らが日本語に訳したものである。Testing のみ 2 つのセッションがあったため、合わせて 1 つのセッションとして扱った。各セッションにつきおおむね 4 本程度の研究論文が含まれる。

回答者全体の傾向(RQ1 に関する分析): 業務に関連すると回答のあった研究カテゴリについても表 3 に示す。総計は従業員数別に層別しない場合に関連があると答えた割合を示す。3 章で説明したように、少なくとも 1 人あたり 3 つのカテゴリを選択しており、その総選択数(少なくとも回答者 16 人×3 カテゴリ)を分母として割合を算出している。その他の列は層別して同様の方法により割合を算出している(各列で合計 100%としている)。

全体でも 36% (26 カテゴリ中 9 カテゴリ) のカテゴリが業務と関連があると答えられていなかった。これは、一部の研究が現時点では一般的な開発と関連が薄いためであると考えられる。例えば Compilers and Emerging Trends (コンパイラ・最新トピック)や Energy and Videos (ソフトウェア省電力化・ビデオの活用)などが該当する。

開発支援を業務とする回答者は若干いたにも関わらず、Empirical (ソフトウェア開発データの分析)などの分析関連のカテゴリは選択されていなかった。このことから、従業員数が 300 人未満の場合(このグループには経験年数が 3 年以上の回答者が含まれる)、または、経験年数が 3 年未満の場合、開発支援の業務においてデータ分析は関連が弱い(優先順位が低い)可能性がある。

上記の分析結果より、RQ1 に対する答えは、「偏りがある(業務と関連が弱い研究カテゴリがいくつか存在する)」となる。

企業規模別の傾向(RQ2 に関する分析): Performance (ソフトウェアの性能)や Collaborative (ソフトウェア開発の人的要因)などについては、従業員数 300 人未満の会社の回答者グループでは、業務との関連が弱かった。なお、300 人以上の企業で上記それぞれに関連があると答えている回答者は 4 人であり、例外的な回答者 1 人が答えたわけではない。表 2 で示したように、回答者が関連する工程に大きな違いがないことから、それらがこの結果に影響している可能性は低い。300 人以上の企業に所属する回答者の経験年数は 3 年未満であるが、300 人未満の企業に所属する回答者は 3 年以上の経験を持つ開発者も含まれていることから、経験年数についても影響している可能性が低い。この結果は、開発対象とするソフトウェア

表 4 研究の有用性および興味 (従業員数別)

従業員数	有用性	興味
300 人未満	2.2	3.6
300 人未満 (経験年数 3 年以上除く)	2.2	3.1
300 人以上 1000 人未満	2.4	2.9
1000 人以上	2.4	2.7

表 5 研究の有用性および興味 (学位別)

最終学位	有用性	興味
学士他	2.3	3.0
修士号	2.3	3.5

の性質の違いや、開発体制の違い(プロジェクトあたりの開発者数の違いなど)が影響している可能性がある。これらは企業の規模との関係が強いと思われる。

API (Application Programming Interface) や Open Source (オープンソースソフトウェア)については、従業員数 300 人未満の企業の回答者については業務との関連が強く、それ以外の企業の回答者の業務とは関連が弱かった。経験年数 3 年未満と 3 年以上両方の開発者がこれらを回答していたことから、経験年数が結果に影響している可能性は高くないと考えられる。また、前述のように関連する工程に大きな違いはないことから、開発対象のソフトウェアの規模が比較的小さく、API や OSS の影響が相対的に大きいことが、このような結果につながった可能性がある。

これらの結果より、関連する(担当する)工程に大きな差がなくても、所属する企業の規模が異なる場合、業務に関連が強い研究カテゴリが異なってくる可能性があるといえる。よって、RQ2 に対する答えは「差異が生じる」となる。

4.3. 研究の有用性などに対する評価

本節では RQ3, RQ4 に関する分析結果を述べる。各研究カテゴリに含まれる研究論文に対して、業務に有用かどうか、研究内容に興味があるかどうかを評価してもらった結果を表 4 に示す。所属する企業の従業員数別に層別したが、規模によって有用性の評価に大きな違いはなかった。評価の平均値は 3 を下回っており、ソフトウェア工学の最先端研究は、一般的なソフトウェア開発に対して業務改善などの即効性は強くない可能性がある。

表 4 において、研究の有用性と比較して、研究に対する興味は低くなかった。ただし、興味の平均値は回答グループによって異なっていた。研究に対する興味は、大学院などで研究に取り組んだ経験があるかどうかの影響している可能性がある。そこで回答者を取

表 6 研究カテゴリ別の有用度の平均値

No.	1	2	7	9	10	12	13	18	19	20	23	24	27
有用度	2.0	3.0	2.5	3.0	2.5	2.5	2.2	1.5	2.3	2.3	2.5	2.3	2.2
回答数	2	3	2	2	2	2	7	2	4	3	2	4	6

表 7 研究カテゴリ別の興味の度合いの平均値

No.	2	7	9	10	12	13	18	19	20	23	24	27
興味	4.0	4.0	3.0	4.5	3.5	2.7	1.5	3.3	2.7	3.0	3.3	2.7
回答数	2	2	2	2	2	7	2	4	3	2	4	6

得学位別に層別した結果を表 5 に示す。修士号取得者のほうが研究に対する興味が平均的に高かった。それぞれの研究論文に対する興味の強弱は、企業規模とは関係が弱く、取得学位との関連が強いと考えられる。これらの結果より、RQ4 に対する答えは「差異は生じない」となる。

カテゴリ別の、研究の有用度に対する評価の平均値を表 6 に、興味の度合いの平均値を表 7 に示す。データ件数が少ないため、ここでは層別せずに平均値を算出した。表では、回答が 2 件以上あったカテゴリのみを示し、平均値が 3 以上となったものを太字で示している。表中の No. は表 3 のものに対応する。個別に見た場合でも、研究の有用性の評価は全体的に低く、Performance (ソフトウェアの性能) と Synthesis (自動プログラム生成) のみ平均値が 3 以上となった。

興味の度合いについては、半数以上の研究カテゴリで平均値が 3 以上となっていた。ただし、Languages (ソフトウェア開発言語) に関しては、有用性、興味の度合いとも 2 を下回っており、全カテゴリで最も低い値となっていた。研究内容は C 言語に関するものが多かったが、回答者と C 言語の関連が弱かった可能性も考えられる。なお、カテゴリ別の有用度と興味の度合いについて相関係数を算出すると 0.47 となった。このことから、有用性と興味の度合いはある程度関連があると考えられるが、回答者は有用性と興味の度合いをある程度独立して評価していると考えられる。

これらの結果より、RQ3 に対する答えは「一部の研究カテゴリを除き、有用であるとはいえない」となる。

5. 研究カテゴリの推薦

いくつかの最先端研究は回答者（開発者）に興味を持たれていた。開発者が興味を持ちそうな研究カテゴリを、あらかじめ推薦システムによって提示することができれば、開発者は読む論文を絞り込むことができる。例えば、推薦システムはある開発者に対し、「業務と関連のある研究カテゴリは保守とテストである。

そのうち、テストの研究に対して興味を持つと予測される」などと提示する。興味を持ちそうな研究を容易に知ることができれば、最先端研究を知ろうとする技術者も増え、そのような技術者の一部が研究者に対して有用性に対するフィードバックを行えば、それぞれの研究の有用性に対する再考察が進む可能性がある。

そこで本章では推薦システムによって、開発者が興味を持ちそうな研究カテゴリを、高い精度で推薦可能かどうかを実験により確かめた。同様の推薦システムは文献[14]でも提案しているが、文献[14]ではソフトウェア開発技術を推薦しているのに対し、本研究ではソフトウェア工学の研究を推薦している。そのため、必ずしも同様の精度が得られるかが明らかではないため実験により精度を確かめた。

実験では、業務に関連するカテゴリの推薦と、興味の度合いの予測を、協調フィルタリングを用いてそれぞれに行った。4.2 節で示したように、業務に関連するカテゴリは開発者によって異なるため、まず関連するかないかを判別予測した。例えば「開発者 A の業務に関連する研究カテゴリは、保守、テストである」と予測する。次に、興味の度合い（5 段階評価）の数値を予測した。例えば、「開発者 A は保守の研究に対する興味は低く、テストの研究に対する興味は高い」などと予測する。

上記の推薦における説明変数は、開発者のプロフィール、すなわち経験年数、関連する開発工程、所属する企業の規模とした。目的変数は関連の強い工程と、研究に対する興味の度合とした。関連する開発工程、所属する企業の規模についてはダミー変数化した。ダミー変数化とは、例えば要求工程が関連している開発者では 1、そうでない開発者では 0 となるような変数を設定することを指す。判別予測、数値予測の精度評価時にはリーブワンアウト法を適用した。リーブワンアウト法では、全データのうちの 1 件を予測対象とし、それ以外を過去（既知）データとすることを、繰り返して行う。

表 8 協調フィルタリングで用いるデータ

	p_1	p_2	...	p_j	t_k
u_1	v_{11}	v_{12}	...	v_{1j}	r_{1k}
u_2	v_{21}	v_{22}	...	v_{2j}	r_{2k}
...
u_i	v_{i1}	v_{i2}	...	v_{ij}	r_{ik}
...
u_m	v_{m1}	v_{m2}	...	v_{mj}	r_{mk}

5.1. 協調フィルタリングの概要

協調フィルタリングは、ユーザにとって好ましい、または役立つと考えられるアイテム(書籍、音楽など)を推薦するための手法として用いられている[4][10][13]. 「協調」とは、ユーザの知識を利用することを意味し、「フィルタリング」とは、大量のアイテムの中から、役立つアイテムだけを選び出して推薦することを意味する. 一般的な協調フィルタリングで推薦を行う場合、各ユーザが各アイテムを(5段階の数値などで)評価していることが前提となる(システムによっては、ユーザがそのアイテムを閲覧したかどうかを評価の代わりに用いることもある). あるユーザが未評価のアイテムが、そのユーザにとって役立つと考えられる場合、そのアイテムを推薦する.

協調フィルタリングの主なアルゴリズムとして、ユーザベース手法とアイテムベース手法の2つがある. ユーザベース手法は、「アイテムの評価(好み)が似たユーザは、どのアイテムに対しても似た評価を行う」と仮定し、推薦を行う. 具体的には、各ユーザの各アイテムに対する評価を要素とするベクトルを、ユーザごとに作成し、そのベクトルのなす角をユーザの類似度とする. そして推薦対象のユーザが未評価で、かつ類似したユーザの評価が高いアイテムを推薦する. ユーザベース手法を用いた推薦システムとして、Resnickら[10]のGroupLensが挙げられる. GroupLensは、Usenetにある多数のニュース記事から、ユーザの好みに合うと予測される記事を選び出して推薦するシステムである.

アイテムベース手法はSarwarら[13]によって提案されたアルゴリズムであり、アイテム間の類似度に基づいて推薦を行う. アイテムベース手法の場合も、各ユーザの各アイテムに対する評価を要素としてベクトルを作成するが、ユーザごとにベクトルを作成するのではなく、アイテムごとに作成し、類似度を計算する. すなわち、「あるグループのユーザに高評価されるアイテムは、類似の性質を持っている」と仮定し、推薦対象のユーザが高い評価を行っているアイテムと類似度の高いアイテムを推薦する.

協調フィルタリングを用いた推薦では、表8のよ

表 9 協調フィルタリングによる判別予測精度

	適合率	再現率	F1 値
ユーザベース	30%	24%	27%
アイテムベース	20%	22%	21%

表 10 協調フィルタリングによる数値予測精度

	絶対誤差平均値
ユーザベース	1.16
アイテムベース	1.23

うな $m \times n$ のマトリクス形式のデータを想定する. $u_i \in \{u_1, u_2, \dots, u_m\}$ は i 番目の開発者を表し、 $p_j \in \{p_1, p_2, \dots, p_j\}$ は開発者のプロフィール(経験年数など)を表す. また、 $t_k \in \{r_{1k}, r_{2k}, \dots, r_{mk}\}$ はある研究カテゴリ k の評価結果を表す.

5.2. 評価尺度

判別予測の評価尺度として、適合率、再現率、F1値を用いた. これらは、推薦システムの評価尺度として広く用いられているものである[5]. 適合率は、「業務に関連がある」と予測されたうち、実際に関連があった割合を示す. 再現率は、全ての関連があった研究カテゴリのうち、「関連がある」と予測されたものの割合を指す. F1値は、適合率と再現率のバランスを考慮して一つの指標としたものである. これらの値が大きいほど、判別予測の精度が高いことを示す.

数値予測の評価尺度には絶対誤差を用いた. 絶対誤差は予測値と実測値(実際の評価値)との差の絶対値であり、値が小さいほど数値予測の精度が高いことを示す.

5.3. 予測結果

判別予測の結果を表9に示す. アイテムベースよりもユーザベースのほうの精度が高かったが、F1値が27%と非常に低い判別予測精度となった. 数値予測の結果を表10に示す. こちらについてもユーザベースの予測精度がわずかではあるが高かった. ただし、絶対誤差の平均値が1を少し超えており、高い精度であるとまではいえなかった.

この結果より、業務との関連が強い研究カテゴリを判別予測することは容易ではないといえる. ただし、データの件数が増えれば予測精度が高まる可能性もある. 実験で用いたデータにおいては、業務に関連する研究カテゴリの絞り込みについては、各開発者で行う必要があるといえる. それらに対する興味の高さの高低については、ある程度は協調フィルタリングにより予測可能であるといえる. また、推薦に用いるア

ルゴリズムにはユーザベースが適しているといえる。

6. 考察

ソフトウェア工学は特に他の工学分野と比べて開発現場と密に関わっているため、理論中心で現場への適用が想定されていないような研究は、あまり推奨されない。ICSEはソフトウェア工学の理論中心というよりも、実験を考慮した研究が比較的多いとみなし、本研究の実験題材として採用した。

研究論文のグルーピングを行う際には、以下の3点をクリアできるよう考慮する必要がある。

- グループの定義が正しい
- 論文が所属するグループが正しい
- 論文数に偏りが出ない

研究論文を分類するために、著者らで新たなグループを考慮することもできたが、論文によってはどのカテゴリに何が属するのかの判断が難しく、論文の数にも偏りが出る可能性があり、上記3点が満たせない可能性があった。各セッションに基づいて論文を分類しても特に違和感はなかったため、やや乱暴ではあるが会議のセッションを研究カテゴリとして用いた。

Zimmermannら[1][8]の研究では、「研究の方向性が正しいか」を聞いており、この観点からは各研究の評価は高かった。それに対し本研究では、実際の業務でソフトウェア工学が有用であるかどうかを重要であると考え、評価してもらった。一部の回答者は業務経験年数が長かったが、有用性の評価は低かった。このことから、業務経験の短さが必ずしも有用性の評価につながっていないと考える。また、研究内容に対する興味については低くなかったため、回答者が内容を理解していないとはいえないと考える。

著者の一人の実務経験に基づく、実務経験が1年ほどあれば、技術の有用性が理解できる程度の知識は習得しており、少なくとも、自分が担当する業務に関しては、研究の有用性に対する判断はある程度正しく行えると考える。ただし、経験年数が少ないと、自分が担当している工程が少ないため、担当した経験の少ない分野については有用性が低くなる可能性がある。特に上流工程に関する研究の評価については、経験年数のより長い被験者を対象に調査をする必要があると考える。

7. おわりに

本研究では、最先端のソフトウェア工学の有用性に対するソフトウェア開発者の評価を、開発者が所属する企業規模に着目して分析した。企業に所属するソフ

トウェア開発者に対し、業務に関係する研究カテゴリを最大で5つ選択してもらい、そのカテゴリに属する研究(1カテゴリにつき4つ程度の研究)が自分の業務に有用かどうか、興味を持ったかどうかを5段階で評価してもらった。その結果、以下の傾向が見られた。

- 業務と関連が弱い研究カテゴリがいくつか存在した。例えば *Compilers and Emerging Trends* (コンパイラ・最新トピック)や *Energy and Videos* (ソフトウェア省電力化・ビデオの活用)などのカテゴリは関連があるとの回答がなかった。
- 所属する企業の規模によって、関連する研究カテゴリに差異が見られた。例えばAPIやOSSは、従業員数300人未満の企業に所属する開発者の業務と関連が強く、それ以外の関連が弱かった。
- *Performance* (ソフトウェアの性能)と *Synthesis* (自動プログラム生成)に関する研究を除き、最先端の研究は業務に有用であるとはいえなかった。
- 開発者が所属する企業の規模によって、研究の有用性及び興味の評価に差異は生じなかった。ただし、興味については開発者の取得学位が影響している可能性がある。
- 業務に関連する研究カテゴリを予測した場合の精度が低かった。ただし、研究に対する興味の度合いについては、ある程度の精度で予測可能であった。

今後さらに分析を行う必要があるが、今回の分析では有用性の評価が高くなく、ソフトウェア工学の最先端研究が中小企業のニーズに応えられているとは必ずしもいえなかった。今後の予定は、回答者を増やすとともに、入社年数の長い回答者を増やして分析結果の信頼性を高めることである。

謝辞 本研究の一部は、文部科学省科学研究補助費(基盤C:課題番号16K00113)による助成を受けた。

参考文献

- [1] Carver, J., Dieste, O., Kraft, N., Lo, D., and Zimmermann, T.: How Practitioners Perceive the Relevance of ESEM Research, *In Proc. of International Symposium on Empirical Software Engineering and Measurement (ESEM)*, no.56, 10pages, 2016.
- [2] 中小企業庁:中小企業の定義について, http://www.chusho.meti.go.jp/pamflet/g_book/h22/teigi.html.
- [3] Estublier, J., Leblang, D., Hoek, A., Conradi, R., Clemm, G., Tichy, W., and Wiborg-Weber, W.:

Impact of software engineering research on the practice of software configuration management, *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol.14, no.4, pp.383-430, 2015.

- [4] Goldberg, D., Nichols, D., Oki, B., and Terry, D.: Using Collaborative Filtering to Weave an Information Tapestry, *Communications of the ACM*, vol.35, no.12, pp.61-70, 1992.
- [5] Herlocker, J., Konstan, J., Terveen, L., and Riedl, J.: Evaluating collaborative filtering recommender systems, *ACM Transactions on Information Systems (TOIS)*, vol.22, No.1, pp.5-53, 2004.
- [6] 情報処理学会 ソフトウェア工学研究会 国際的研究活動活性化 WG : ICSE2016 勉強会 , <https://sites.google.com/site/sereadings/icse2016>.
- [7] 経済産業省:平成 27 年特定サービス産業実態調査, 2016.
- [8] Lo, D., Nagappan, N., and Zimmermann, T.: How practitioners perceive the relevance of software engineering research, In Proc. of Joint Meeting on Foundations of Software Engineering (ESEC/FSE), pp. 415-425, 2015.
- [9] Osterweil, L., Ghezzi, C., Kramer, J., and Wolf A.: Determining the Impact of Software Engineering Research on Practice, *Computer*, vol.41, no.3, pp.39-49, 2008.
- [10] Resnick, P. Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J.: GroupLens: An Open Architecture for Collaborative Filtering of Netnews, In Proc. of Conference on Computer Supported Cooperative Work (CSCW), pp.175-186, 1994.
- [11] Rubin, J., and Rinard, M.: The Challenges of Staying Together While Moving Fast: An Exploratory Study, In Proc. of International Conference on Software Engineering (ICSE), pp.982-993, 2016.
- [12] Ryder, B., Soffa, M., and Burnett, M.: The impact of software engineering research on modern programming languages, *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol.14, no.4, pp.431-477, 2005.
- [13] Sarwar, B., Karypis, G., Konstan, J., and Riedl, J.: Item-Based Collaborative Filtering Recommendation Algorithms, In Proc. of International World Wide Web Conference (WWW), pp.285-295, 2001.
- [14] 秋永知宏, 大杉直樹, 柿元健, 角田雅照, 門田暁人, 松本健一:協調フィルタリングに基づくソフトウェア開発技術の推薦, 電子情報通信学会技術報告, ソフトウェアサイエンス研究会, SS2005-13, pp.7-13, June 2005.

実験参加協力をお願い

我々は,ソフトウェア開発企業で開発業務に従事したことがある方で実験参加の協力をしていただける方を募集しています. 詳細は以下 google フォームのサイトに記載しております. お問い合わせは, 本研究の著者までお願いします.

<https://goo.gl/forms/n5UKrffJRAfkKFXz1>