# Is Cutting-Edge Software Engineering Attractive for Developers in SMEs?

Yukasa Murakami, Masateru Tsunoda
Graduate School of Science and Engineering Research
Kindai University
Higashiosaka, Japan
m.yukasa@gmail.com, tsunoda@info.kindai.ac.jp

***Abstract*—To set the goal of software engineering research, we should aware needs of the majority of developers. In Japan, the majority is developers who belong to companies where number of employees is smaller than 300. However, it is not clear to what extent cutting-edge software engineering satisfies the needs of the majority. So, we analyzed whether cutting-edge software engineering satisfies the needs of developers who belong to small and medium-sized enterprises, or not. We asked 16 software developers who belong to software development companies about effectiveness of cutting-edge researches to their work. In the analysis, we stratified data based on the size of their company. The analysis results suggested that when size of companies to which subjects belong was different, research categories related to subjects' work were also different. However, sizes of companies did not affect evaluation to the effectiveness and the interests of the researches. Additionally, we predicted categories of researches which relate to each developer's business (i.e., discrimination), and the evaluation of developers (evaluated by five grades) to each research, using collaborative filtering. As a result, when we predicted research categories which related to subjects' work, the accuracy was low (F1 score was 27%). In contrast, when we predicted the interestingness to the researches, the accuracy was moderate (Average of absolute error was 1.16).**

**Keywords—investigation; questionnaire; software developers; recommendation**

## I. Introduction

Recently, software is one of infrastructures of society, and embedded software and enterprise software are indispensable for social activities. Therefore, high reliability is required for software. On the other hand, large software is required to be developed in short term. To enhance the probability of success of such projects, assistances based on software engineering techniques are needed.

When we cope with studies of the assistances based on software engineering, we should be aware of needs of majority of software developers. In Japan, a software development company is legally regarded as a small and medium-sized enterprise (SME), when the number of employees is smaller than 300 or stated capital is smaller than 300 million yen [15]. The ratio of software development companies where number of employees is smaller than 300 is 99% in Japan. Additionally, the ratio of software developers who are employed by the companies is 69% [8]. Therefore, the majority of software developers in Japan belong to such companies with less than 300 employees. There are also many SMEs in the United States of America [16] and EU [4].

It is not clear to what extent cutting-edge software engineering satisfies the needs of the majority of software developers. Some studies investigated the effectiveness or correctness of direction of software engineering for practitioners [9][12]. However, subjects of their investigation were developers who belong to large companies such as Facebook or Microsoft. In small and medium-sized companies, circumstances such as number of software project members and size of software are considered to be different from large companies. So, when size of a company is different, needs of developers may be also different.

In this study, we analyze whether cutting-edge software engineering satisfies the needs of developers who belong to small and medium-sized enterprises, or not. It is important for cutting-edge software engineering to satisfy the needs of large companies. Similarly, cutting-edge software engineering should consider the needs of small and medium-sized enterprises. In our study, subjects who belong to software development companies evaluated the effectiveness of cutting-edge software engineering for their business. After that, we stratified the data based on size of their companies, and analyzed it. In addition, we predicted categories of researches which relate to each developer's business, and the evaluation of developers to each research, using collaborative filtering. The purpose of the prediction is to support the evaluation of research by developers.

## II. Related Work

Impact Project [10] analyzed the influence of software engineering research to software development. For example, they analyzed how software engineering contributes to evolution of programming language and software configuration managements [3][13]. As a result, they concluded that software engineering research affects actual software development process. The result suggests effectiveness of software engineering. However, it is not clear that whether cutting-edge software engineering is also effective or not, and ratio of effective cutting-edge studies (when some studies are not effective).

TABLE I. SUMMERY OF SUBJECTS

| Number of employees | Number of subjects | Years of experience < 3 | Average years of experience | Master's degree |
|---|---|---|---|---|
| Less than 300 | 6 | 4 | 6.7 | 2 |
| 300 to 999 | 5 | 5 | 1.2 | 2 |
| More than 999 | 5 | 5 | 1.2 | 1 |

TABLE II. DEVELOPMENT PROCESS IN WHICH SUBJECTS ARE INVOLVED.

| Number of employees | Requirement analysis | Basic design | Detailed design | Testing | Coding | Development support | Project management |
|---|---|---|---|---|---|---|---|
| Less than 300 | 33% | 67% | 67% | 100% | 83% | 33% | 17% |
| Less than 300 (Years of exp. < 3) | 0% | 50% | 50% | 100% | 75% | 0% | 0% |
| 300 to 999 | 20% | 60% | 60% | 80% | 80% | 20% | 0% |
| More than 999 | 20% | 60% | 60% | 100% | 80% | 0% | 20% |

Rubin et al. [12] discussed directions of software engineering research. They interviewed developers who belong to frontier software companies such as Google and Facebook. They clarified issues faced by the developers, and based on that, they suggest needs for researches which help developers to access useful information quickly. However, they do not clarify needs of developers who do not belong to frontier software companies.

There are two studies which are similar to our study [2][9]. The studies were conducted by the same research group. In the studies, subjects who belong to software development companies read summaries (about one or two lines) of papers of international conferences on software engineering. After that, subjects evaluated value of the researches by five grades. As a result, about 70% of studies are regarded as valuable.

Our investigation procedure is based on the existing studies [2][9]. However, our study has three major differences from them. The first one is viewpoint of evaluation to each research. In this study, subjects evaluated researches based on usefulness on their own businesses. This is the most important difference. The second one is that our analysis focused on size of software companies to which subjects belong. In the existing studies, major subjects belong to large software companies such as Microsoft. The third one is that researches were evaluated based on concise explanatory documents (They are longer than the two-line abstracts). As a result, the evaluation to cutting-edge studies was not high, in contrast to the existing studies.

## III. DATA COLLECTION

We collected data from developers who belong to software companies, using a questionnaire (Google Forms). Subjects answered questions such as effectiveness of researches based on the following procedure:

1. Choose categories of researches which relate to their work (at least three categories, and at most five categories).
2. Read the concise explanatory documents of each research which belongs to the top three categories selected in step 1.
3. Subjects evaluate papers based on usefulness for their work by five grades (1: lowest and 5: highest). In addition, they evaluate papers based on their interest.

We decided categories of the researches and assignment of papers to the categories, based on ICSE (International Conference on Software Engineering) 2016. ICSE is the top conference on software engineering field. In more detail, we set each session in the conference as each category of researches. The detail of the categories is explained in section IV.B. Each category (i.e., session) includes about four papers. To shorten answer time of subjects, subjects read summaries of researches [7], instead of reading papers of them directly. Each summery consists of two or three presentation slides, and it is relatively easy to understand.

Additionally, subjects answered the following questions about themselves:

- Academic degree (e.g., bachelor or master)
- Years of work experience
- Number of employees in subjects' companies (less than 300, 300 to 999 employees, more than 999)
- Development process in which subjects are involved.

## IV. ANALYSIS OF ANSWERS

To clarify the purpose of the analysis, we set the following research questions.

- RQ1: Do research categories relate to development work biasedly?
- RQ2: When size of subjects' company is different, research categories which relate to their works is also different?
- RQ3: Is results of cutting-edge researches effective for subjects' works?
- RQ4: When size of subjects' company is different, effectiveness of researches to their work is also different?

TABLE III. RATIO OF RESEARCH CATEGORIES RELATED TO SUBJECTS' WORK.

| No. | Research category | # of papers | Total | Less than 300 | Less than 300 (Years of exp. < 3) | 300 to 999 | More than 999 |
|---|---|---|---|---|---|---|---|
| 1 | Android (Software on Android) | 4 | 2% | | | | 8% |
| 2 | Performance (Performance of software) | 4 | 4% | | | 7% | 8% |
| 3 | Empirical (Software data analysis) | 3 | | | | | |
| 4 | Symbolic Execution (Software analysis by symbolic execution) | 4 | | | | | |
| 5 | Compilers and Emerging Trends | 4 | | | | | |
| 6 | Energy and Videos (Energy saving of software, and utilizing videos) | 4 | | | | | |
| 7 | Open Source (Open source software) | 2 | 4% | 11% | 8% | | |
| 8 | Defect Prediction (Software defect prediction) | 4 | | | | | |
| 9 | Synthesis (Automatic code generation) | 4 | 4% | | | 7% | 8% |
| 10 | API (Application Programming Interface) | 3 | 4% | 11% | 8% | | |
| 11 | Code Smells (Coding methodologies) | 4 | 2% | | | | 8% |
| 12 | Architecture (Software architecture) | 3 | 4% | 6% | | 7% | |
| 13 | Testing (Software testing) | 8 | 20% | 17% | 17% | 21% | 23% |
| 15 | Effort Estimation and Search (Software effort estimation) | 3 | 2% | 6% | 8% | | |
| 16 | Product Lines (Reuse software by product lines) | 4 | | | | | |
| 17 | Repair and Model Synthesis (Automatic software repair) | 4 | | | | | |
| 18 | Languages (Programming languages) | 4 | 4% | 6% | 8% | 7% | |
| 19 | Debugging (Debugging software) | 4 | 9% | 11% | 8% | 7% | 8% |
| 20 | Requirements (Software requirements) | 4 | 7% | 6% | 8% | 7% | 8% |
| 21 | Dynamic Analysis (Dynamic analysis of software) | 4 | 2% | | | 7% | |
| 22 | Security | 4 | 2% | 6% | | | |
| 23 | Collaborative (Human factor of software development) | 4 | 4% | | | 7% | 8% |
| 24 | Software Quality | 3 | 9% | 6% | 8% | 7% | 15% |
| 25 | Program Analysis | 4 | | | | | |
| 26 | Concurrency (Concurrent execution of software) | 4 | | | | | |
| 27 | Maintenance (Software maintenance) | 4 | 13% | 17% | 25% | 14% | 8% |

### A. *Overview of Subjects*

Subjects of this study is 16 software developers who belong to software development companies. Table I is summary of subjects, stratified by size of their companies. On the group where number of employees was smaller than 300 (i.e., first row of the table), there were two developers whose years of work experience was more than three years. On other groups, years of work experience was less than three years. To suppress the influence of years of work experience, we also show the analysis results, excluding subjects whose years of work experience was more than three years from the analysis.

**Development process of subjects**: Table II shows development process in which subjects are involved, stratified by size of their companies. Most of subjects were involved in coding and testing. About half of subjects were involved in design. Subjects were not involved in requirement analysis and development support, when company size was small and years of work experience was low (i.e., second row in the table). In other cases, some subjects were involved in the process.

### B. *Research Categories Relating to Business*

Table III shows research categories used in the analysis. We used sessions of ICSE conference as research categories. We added explanations to each categories as shown in the parenthesis. There were two testing sections in the conference, and we merged the sessions in the analysis. Each session included about four research papers.

**Analysis of all subjects (relating to RQ1)**: Table III also shows research categories which were regarded to relate subjects' work in subjects' answers. In the table, "total" column indicates ratio of answers which recognized the relationship without stratification by company size. As explained in section 3, each subjects select categories, and therefore the denominator is the total number of selected categories (e.g., multiply 16 subjects by 3...5 categories). Other columns denotes the ratio, stratified by company size. Total of the ratio is 100% on each column.

On 36% (9 out of 26 categories) of the categories, the relationship to subjects' work were not observed. This is because some categories do not relate to ordinary software development very much. For example, "compilers and emerging trends" and "energy and videos" are not considered to relate to ordinary software development.

Analysis categories such as "empirical" category were not selected by subjects, although some subjects are involved in development support. So, data analysis may not relate to subjects' work very much (i.e., low priority) when the number of employees was smaller than 300 (Note that this group includes subjects whose years of work experience is more than three years).

Therefore, the answer to RQ1 is "Yes (There are some research categories which do not relate to subjects' work very much)."

**Analysis stratified by company size (relating to RQ2)**: "Performance" and "collaborative" categories did not relate to subjects' work explicitly (i.e., the categories were not selected by subjects), when the number of employees was less than 300. In contrast, some subjects selected the categories in other cases

(Note that the number of subjects who selected the categories was four, and therefore it is not ignorable). As shown in Table II, development process in which subjects are involved was not very different from each subject. So, the difference of the process is not considered to affect the result very much. On the group where number of employees was smaller than 300, there were developers whose years of work experience was more than three years. That is, it is not probable that the lack of the experience affected the result. We think that the result arose from the differences of characteristics of target software and project organization (e.g., number of developers in a project) between small companies and other companies.

"API" and "open source" categories related to subjects' work, when the number of employees was less than 300. In other cases, the strength of the relationships were weak (i.e., the categories were not selected by subjects). The categories were selected by subjects belonging to small companies, regardless of their years of work experience. So, the work experience is not considered to affect the result. Also, the process in which subjects are involved is not very different among subjects, as explained before. When software size is relatively small (i.e., in small companies), the influence of API and OSS to software may be relatively large, and this may affect the analysis results.

The results suggest that when the process in which subjects are involved is not very different, but size of companies to which subjects belong is different, research categories related to subjects' work may be also different. So, the answer of RQ2 is "Yes." evaluation to research

### C. Evaluation to Research

This section explains analysis results relating to RQ3 and RQ4. Table IV shows evaluation of researches included in each category. They were evaluated based on effectiveness to subjects' work and interestingness to subjects separately. We stratified data based on size of companies to which subjects belong. Evaluation of the effectiveness was not very different when size of companies was different. The average of evaluation is less than 3. So, cutting-edge software engineering may not be explicitly effective for improvements of subjects' work.

In Table IV, the interestingness for researches was not low, compared with the effectiveness. However, average of the interestingness was different among groups in the table. We assumed that the interestingness is different, when subjects have engaged in study in graduate school. So, we stratified subjects based on their academic degree, as shown in Table V. On average, the interestingness was relatively high, when subjects' academic degree was master. Therefore, we consider that the interestingness for researches was not affected by size of companies, but by subjects' academic degree. From the result, we answer to RQ4 as "No."

Table VI shows average of effectiveness of each research category, and Table VII shows average of the interestingness of each category. We did not stratified the data because the number of data point of each category was small in the tables. In the table, we removed categories in which the number of data points was smaller than 2. Bold face in the table indicates the average was larger than 3. In Table VI and VII, "No." refers to categories in Table III. Even when we focused on each research category, evaluation of the effectiveness was low in most cases. Only average of the effectiveness of "performance" and "synthesis" was higher than 3.

TABLE IV. EVALUATION OF SUBJECTS STRATIFIED BY SIZE OF COMPANIES.

| Number of employees | Effect | Interest |
|---|---|---|
| Less than 300 | 2.2 | 3.6 |
| Less than 300 (Years of exp. < 3) | 2.2 | 3.1 |
| 300 to 999 | 2.4 | 2.9 |
| More than 999 | 2.4 | 2.7 |

TABLE V. EVALUATION OF SUBJECTS STRATIFIED BY ACADEMIC DEGREE.

| Academic degree | Effect | Interest |
|---|---|---|
| Bachelor and others | 2.3 | 3.0 |
| Master | 2.3 | 3.5 |

In the tables, average of the interestingness was larger than 3 on more than half of research categories. Correlation coefficients between the interestingness and the effectiveness was 0.47. That is, although the effectiveness relates to the interestingness to some extent, the effectiveness does not strongly depend on the interestingness. On "language" category, both average of the interestingness and the effectiveness was smaller than 2, and it was the lowest in the categories. Topics of the researches included in the category were mainly C language. C language may not relate to subjects' work very much.

So, the answer of RQ3 is "It is not very effective, except for some research categories."

## V. RECOMMENDATION OF RESEARCH CATEGORY

### A. Overview of Experiment

Subjects were interested in some research categories. If recommender system suggests research categories in which developers are likely to be interested in advance, it will help developers search attractive researches. For example, recommender system suggests "Research categories which relate to your work are testing and maintenance. You may interested in testing researches." to a developer. If developers know attractive researches easily, more developers will know cutting-edge software engineering, and some of them give feedback to researches. It is expected to promote reconsidering the effectiveness of each research.

To clarify whether the recommender system make accurate recommendation or not, we predicted research categories in which developers are interested, and evaluated the prediction accuracy. Although one of authors proposed similar recommender system in [1], independent variables used for prediction are different. Study [1] used evaluation of each item as independent variables, this study uses profile of developers explained later. In contrast to study [1], research categories are recommended for developers not to read research papers by our method.

TABLE VI. AVERAGE OF EFFECTIVENESS OF EACH CATEGORY.

| No. | 1 | 2 | 7 | 9 | 10 | 12 | 13 | 18 | 19 | 20 | 23 | 24 | 27 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Effect | 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 |
| # of answers | 2 | 3 | 2 | 2 | 2 | 2 | 7 | 2 | 4 | 3 | 2 | 4 | 6 |

TABLE VII. AVERAGE OF USEFULNESS OF EACH CATEGORY.

| No. | 2 | 7 | 9 | 10 | 12 | 13 | 18 | 19 | 20 | 23 | 24 | 27 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Interest | **4.0** | **4.0** | **3.0** | **4.5** | **3.5** | 2.7 | 1.5 | **3.3** | 2.7 | **3.0** | **3.3** | 2.7 |
| # of answers | 2 | 2 | 2 | 2 | 2 | 7 | 2 | 4 | 3 | 2 | 4 | 6 |

In the experiment, the research categories and degree of the interest to the categories were predicted, using collaborative filtering. As shown in section IV.B, some categories did not relate to subjects' work very much. Therefore, prediction (discrimination) of research categories is needed as the first step. For example, the prediction is "Research categories in which developer A is interested are maintenance and testing." As the next step, we predicted the degree of the interest (five grades). For instance, the prediction is "Interest of developer A to maintenance is 2, and the interest to testing is 5."

In the prediction, developers' profile was used as independent variables. That is, years of work experience, development process in which developers are involved, and size of their companies were used. The development process and the size of the companies were transformed into dummy variables, because they are categorical variables. We applied leave-one-out cross validation to evaluate the prediction accuracy.

As evaluation criteria of discrimination (i.e., prediction of research categories), we used precision, recall, and F1 score. They are widely used to evaluate performance of recommender system [6]. When the values are large, the prediction is accurate. To evaluate prediction of the degree of the interest, we used absolute error. Absolute error is absolute value of difference between predicted value and actual value (actual evaluation to researches by subjects). When the value is small, the prediction is accurate.

### B. *Collaborative Filtering*

Collaborative filtering is used to recommend preferable or useful items such as books and music [5][11][14]. There are two major recommendation methods of collaborative filtering. The first one is user based method. The user based method assumes "When users are similar, their preference is also similar." First, using values of independent variables, cosine similarity is calculated. Next, when the similarity is high, a value of dependent variable is predicted based on other similar users. User based method was proposed to recommend news articles [11]. The other major recommendation methods is item based method [14]. The method recommends items based on similarity of items. Although the method calculates cosine similarity, the direction is different. That is, user based method compares each rows, and item based method compares each columns.

Collaborative filtering uses $m \times n$ matrix as shown in Table VIII. $u_i \in \{u_1, u_2, ..., u_m\}$ denotes $i$-th developers. $p_j \in \{p_1, p_2, ..., p_j\}$ is profile of developers such as years of the experiments. $t_k \in \{r_{1k}, r_{2k}, ..., r_{mk}\}$ is evaluation to research categories.

### C. *Results of Prediction*

Table IX shows prediction accuracy of the research categories. The accuracy of the user based prediction was higher than the item based prediction. However, F1 score of the prediction was 27%, and hence it was very low accuracy. Prediction accuracy of the degree of the interest is shown in Table X. User based prediction was also slightly higher. Average of the absolute error was slightly larger than 1, and therefore the accuracy was not very high.

From the result, it is not easy to predict research categories in which developers are interested. Therefore, it may be better for developers to select the research categories without recommender system. When the number of data points is increased, the prediction accuracy may be improved. In contrast, the degree of the interest can be predicted by collaborative filtering to some extent. To predict research categories and the degree of the interest, the user based prediction is more proper than item based prediction.

## VI. DISCUSSION

Software engineering strongly relates to actual development field, compared to other engineering areas. So, research of software engineering generally consider application

TABLE VIII. DATA MATRIC USED IN COLLABORATIVE FILTERING.

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

TABLE IX. PREDICTION ACCURACY OF RESEARCH CATEGORIES.

| Prediction method | Precision | Recall | F1 score |
|---|---|---|---|
| User based | 30% | 24% | 27% |
| Item based | 20% | 22% | 21% |

TABLE X. PREDICTION ACCURACY OF SUBJECTS' EVALUATION.

| Prediction method | Average of absolute error |
|---|---|
| User based | 1.16 |
| Item based | 1.23 |

to actual development. On ICSE, theoretical research is not dominant, compared with ASE and FSE. So, we picked up researches presented in ICSE.

When research papers are categorized, we should satisfy the following points:

- The definitions of the categories are correct.
- Classification of the papers based on the categories is correct.
- The number of papers included in each category is not biased very much.

Instead of using the sessions of the conference, we can make new categories to classify the papers. However, it is not easy to classify the papers, and therefore, it is difficult to satisfy the above points. We checked papers included in the sessions, and the classification based on the session did not seem incompatible. So, we used the session to classify the papers, although it is a bit rough approach.

Existing studies [2][9] asked correctness of directions of researches (i.e., general values of researches), and the researches were highly evaluated on the view point. In contrast, we focused on usefulness of research to each subject's work, because the usefulness is more important for the subjects than the general values. Although years of work experience was long on some subjects, their evaluation to the usefulness was low. So, we consider that lack of the experience did not affect such low evaluation very much. We regard that subjects understand researches to some extent, because subjects' interest to researches was not low.

Based on the experience of one of us, when years of work experience is larger than one year, developers have knowledge to understand usefulness of technologies. So, they can properly judge usefulness of researches to some extent, when the researches relate to developers' work.

However, when years of the experience is low, knowledge of some development process in which subjects are involved may be relatively low, and it could affect the evaluation. Especially, to evaluate researches about early development phase more properly, we should add subjects who have more years of the experience.

## VII. Conclusions

In this study, we analyzed software developers' evaluation to effectiveness of cutting-edge software engineering for their work, focusing on size of companies to which the developers belong. First, the developers who work in software companies selected three to five research categories which relate to their work. Next, they evaluated researches included in the categories (about four researches were included in each category) by five grades, considering effectiveness of the researches to their work and interestingness. We analyzed the data, and observed the followings:

- Relationships of some research categories to developers' work were weak. For example, "compilers and emerging trends" and "energy and videos" were note selected by subjects.
- When size of companies to which subjects belong was different, research categories related to subjects' work were also different. For example, "API" and OSS" were selected only when the number of employees were smaller than 300.
- Except for "performance" and "synthesis," cutting-edge researches were not evaluated as effective to subjects' work.
- Size of companies did not affect evaluation to the effectiveness and the interestingness of the researches. Academic degree may affect the interestingness.
- When we predicted research categories which related to subjects' work, the accuracy was low. In contrast, when we predicted the interestingness to the researches, the accuracy was moderate.

To conclude the analysis, we need more subjects. In this case study, the effectiveness was not high, and we do not think that cutting-edge software engineering does not satisfy needs of developers in SMEs (small and medium-sized enterprises). Our future work is to increase the number of subjects. Especially, we should recruit subjects whose years of work experience is long.

## Acknowledgment

This research was partially supported by Japan Society for the Promotion of Science(JSPS) [Grants-in-Aid for Scientific Research (C) and (A) (No.16K00113 and No.17H00731)].

## References

[1] T. Akinaga, N. Ohsugi, M. Tsunoda, T. Kakimoto, A. Monden and K. Matsumoto, "Recommendation of software technologies based on collaborative filtering," In Proc. of Asia-Pacific Software Engineering Conference (APSEC), pp. 209-214, 2005.

[2] J. Carver, O. Dieste, N. Kraft, D. Lo, and T. Zimmermann, "How Practitioners Perceive the Relevance of ESEM Research," In Proc. of International Symposium on Empirical Software Engineering and Measurement (ESEM), no.56, 10 pages, 2016.

[3] J. Estublier, D. Leblang, A. Hoek, R. Conradi, G. Clemm, W. Tichy, and E. Wiborg-Weber, "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] European Commission, What is an SME? http://ec.europa.eu/growth /smes/business-friendly-environment/sme-definition_en

[5] D. Goldberg, D. Nichols, B. Oki, and D. Terry, "Using Collaborative Filtering to Weave an Information Tapestry," Communications of the ACM, vol.35, no.12, pp.61-70, 1992.

[6] J. Herlocker, J. Konstan, L. Terveen, and J. Riedl, "Evaluating collaborative filtering recommender systems," ACM Transactions on Information Systems (TOIS), vol.22, No.1, pp.5-53, 2004.

[7] IPSJ/SIGSE Working Group of Activation of International research activities, ICSE2016 study meeting, https://sites.google.com/site/ sereadings/icse2016 (in Japanese).

[8] Ministry of Economy, Trade and Industry, Japan, Survey of Selected Service Industries in 2015，2016 (in Japanese).

[9] D. Lo, N. Nagappan, and T. Zimmermann, "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.

[10] L. Osterweil, C. Ghezzi, J. Kramer, and A. Wolf, “Determining the Impact of Software Engineering Research on Practice,” Computer, vol.41, no.3, pp.39-49, 2008.

[11] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, “GroupLens: An Open Architecture for Collaborative Filtering of Netnews,” In Proc. of Conference on Computer Supported Cooperative Work (CSCW), pp.175-186, 1994.

[12] J. Rubin, and M. Rinard, “The Challenges of Staying Together While Moving Fast: An Exploratory Study,” In Proc. of International Conference on Software Engineering (ICSE), pp.982-993, 2016.

[13] B. Ryder, M. Soffa, and M. Burnett, “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.

[14] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, “Item-Based Collaborative Filtering Recommendation Algorithms,” In Proc. of International World Wide Web Conference (WWW), pp.285-295, 2001.

[15] The Small and Medium Enterprise Agency, Japan, Definition of small and medium-sized enterprises, http://www.chusho.meti.go.jp/pamflet/g_book/h22/teigi.html (in Japanese).

[16] United States Trade Representative, Small and Medium-Sized Enterprises, https://ustr.gov/trade-agreements/free-trade-agreements/trans-pacific-partnership/tpp-chapter-chapter-negotiating-8