

ソーシャルコーディングプラットフォーム上の サービス連携のためのメタボット

玉田 春昭^{1,a)}

概要：今日、ソフトウェア開発を支援するサービスは数多く利用されている。その一方で、開発者に知られておらず、利用されていないサービスも数多く存在する。そこで、セレンディビティに着目した情報推薦に着目し、未知かつ有益なサービスを開発者に推薦するボットの構築を目指す。これにより、今まで曖昧であったサービス利用の根拠や、ボットからの定期的な推薦による利用サービスの見直しが期待できる。

Towards Meta Bot Service for Integrating Services on Social Coding Platform

HARUAKI TAMADA^{1,a)}

1. はじめに

高品質なソフトウェアを素早く開発するために様々な支援ツールやサービスが提案されている。近年は ChatOps, DevOps と呼ばれる概念が浸透し、様々なボットがソフトウェア開発を支援するようになってきている。また、ソーシャルコーディングのプラットフォーム（以降、SCP）として GitHub^{*1} や GitLab^{*2}, Bitbucket^{*3} などが広く使われている。SCP では、複数の外部サービスを利用しているプロジェクトも多く存在する。それらのサービスは、プルリクエスト（以降、PR）の発行や、リポジトリへの push をトリガとして実行され、結果を PR などのコメントとして提示する。すなわち、事前に決められたトリガにより、開発者がサービスを起動し結果を受け取るようになってきている（図 1）。この一連の動作は、SCP 上にボットが常駐しているものと見做せる。

このような使い方のサービスは、例えば、テストカバレッジを測定するサービス^{*4} や継続的インテグレーション

サービス（CI）^{*5}、また、コード品質測定サービス^{*6}などが挙げられる。これらのサービスを開発者が選択、設定することで、自動的にプロジェクトの測定が行われ、結果が得られる。しかし問題は、開発者にとって未知のサービスが数多く存在することである。また、同じ目的のサービスであっても、一般的に結果の提示方法が違っており、開発者に合うサービスの発見は困難である。

そこで本稿では、開発者にとって未知かつ有益なサービスを見つけ出す方法について議論する。以降、第 2 節で提案手法の利用シナリオについて述べ、続く第 3 節で提案手法を説明する。第 4 節では、考慮すべき事柄について議論

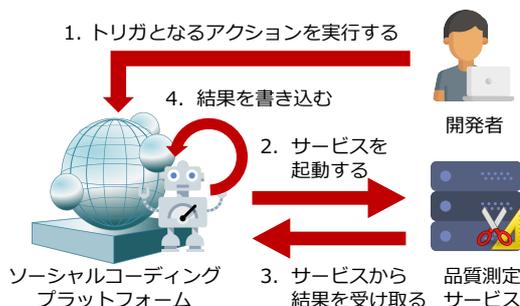


図 1 プラットフォーム上に常駐するボットの振舞

¹ 京都産業大学
Motoyama, Kamigamo, Kita-ku, Kyoto, 630-8555, Japan

a) tamada@cc.kyoto-su.ac.jp

*1 <https://github.com/>

*2 <https://about.gitlab.com/>

*3 <https://bitbucket.org/>

*4 Coveralls <https://coveralls.io/>

*5 Travis-CI <https://travis-ci.org/>

*6 Codebeat <https://codebeat.co/>

する。最後に第5節で本稿をまとめる。なお、本稿では、SCP 上におけるボットについての議論であり、Slack*⁷ などのチャットや個人の PC 上でのボットは対象外とする。

2. 提案手法の利用シナリオ

例えば、CI サービスとして、Travis CI, Bamboo*⁸, CircleCI*⁹, の3つを考える。どのサービスも SCP と連携し、PR の発行をトリガとしてサービスが起動する。しかし、行うことは同じであるものの、対応している言語や結果として提示する内容にサービスごとの特徴が現れている。

ある開発者 A があるプロジェクト P で Bamboo を利用しているとする。しかし、Bamboo を Travis CI に置き換えても支障がない場合が多い。Bamboo を利用する強い理由がないとも言える。そのような状況の中、Bamboo よりも適当な CI サービスが新たにリリースされたとしても、それを見直すタイミングは、開発者の日頃の調査に依存してしまう。そこで、開発者にサービスを定期的に推薦するボットを考える。これにより、利用サービスを見直すタイミングと、当該サービスを利用する理由を与えられると期待できる。推薦対象となるサービスは、開発者にとって未知かつ有益であることが望ましい。既知かつ有益であれば、すでに利用しているか、より有益な同様のサービスを利用していると考えられる。未知で無益なものしか推薦しなければ、提案ボット自体が利用されなくなるためである。

3. 提案手法

本稿の目的は、開発者にとって未知かつ有益なサービスを見つけ出すことである。しかし、開発者ごとに未知なサービス、有益なサービスは異なる。

一方で、セレンディピティに着目した情報推薦が研究されている [1]。ここでは、既知の広く知られているアイテムを推薦結果から削除することで、セレンディピティなアイテムが残る可能性を高める方法 [2] や、イノベータ理論 [3] に基づいて、イノベータが評価するアイテムを推薦する方法などがある [4]。また、ユーザとの積極的なインタラクションを発生させ、ユーザが能動的にセレンディピティなアイテムを発見する方法も提案されている [5]。

そこで、セレンディピティに着目した情報推薦手法を用いて、未知かつ有益なサービスを推薦することを目指す。そのために、既存のサービスの情報を取得し、類似サービスが実行された時に、開発者にとって未知かつ有益であろうサービスを推薦する。

4. 議論

開発者にとって未知かつ有益のサービスが見つかったと

しても、提示方法についても検討する必要がある。以下に検討が必要な各項目を挙げ、議論する。

推薦内容 単純にサービスを推薦するだけでなく、そのサービスを利用することによるメリットを与える必要がある。同系統のサービスをすでに利用している場合は、乗り換える理由を提示する必要がある。

推薦タイミング 本稿で対象としているボットは SCP 上のボットである。リアルタイム性は求められない反面、開発者のアクション (push, PR, issue の発行など) に応じた反応が可能となる。また、他のサービスの実行結果にも反応できる。

推薦頻度 頻繁な推薦は開発の妨げになる他、ボットのメッセージを確認する煩わしさから提案ボット自体を除外されてしまう可能性がある。適切な推薦頻度をしっかりと計測、評価する必要がある。

信頼性 新たなサービスを推薦したとしても、利用中のサービスの方が適していた場合、提案ボットの信頼性が損なわれる恐れがある。間違った推薦を行っても許されるようなインターフェースが重要である。また、新たなサービスへの移行や元のサービスに戻ることに手軽さも信頼性の維持には必要であろう。そのためには、ボットが既存サービスの設定を推薦サービスの設定に変換する機構を用意する必要がある。

5. まとめ

本稿では、開発者にとって未知かつ有益のサービスを推薦する手法について議論した。未知のサービスは開発者ごとに異なり、有益かどうかは、開発者、プロジェクトによっても異なる。そこで、セレンディピティに着目した情報推薦を用いて、未知かつ有益のサービスを推薦することを目指す。ただし、推薦内容やタイミング、頻度や推薦のインターフェースなどを考慮する必要があることを確認した。

謝辞 本研究の一部は JSPS 科研費 17K00500, 17H00731 の助成を受けた。

参考文献

- [1] 奥 健太: セレンディピティ指向情報推薦の研究動向, 日本知能情報ファジィ学会誌, Vol. 25, No. 1, pp. 2-10 (2013).
- [2] Sarwar, B., Karypis, G., Konstan, J. and Reidl, J.: Item-based collaborative filtering recommendation algorithms, *Proc. the 10th International Conference on world wide web*, pp. 285-295 (2001).
- [3] Rogers, E. M.: *Diffusion of Innovations, 5th Edition*, Free Press (2003).
- [4] Kawamae, N.: Serendipitous recommendations via innovators, *Proc. the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 218-225 (2010).
- [5] Oku, K. and Hattori, F.: Fusion-based Recommender System for Serendipity-Oriented Recommendations, *Journal of Japan Society for Fuzzy Theory and Intelligent Informatics*, Vol. 25, No. 1, pp. 524-539 (2013).

*⁷ <https://www.slack.com/>

*⁸ <https://ja.atlassian.com/software/bamboo/>

*⁹ <https://circleci.com/>