

ソーシャルコーディングにおける有益提案の抽出支援システムの検討 Support System for Extraction of Significant Contributions in Social Coding

藤原裕貴[†]
Yuuki Fujiwara

乃村能成[†]
Yoshinari Nomura

1. はじめに

ソフトウェア開発において、ソーシャルコーディングと呼ばれる手法が広がっている。ソーシャルコーディングの手法を取り入れた代表的なサービスに GitHub[1]がある。

ソーシャルコーディングで開発が進められるプロジェクトは公開されているため、プロジェクト内外のさまざまなユーザからソースコードの改善が提案される。提案の中には、プロジェクトに取り込むべきではないものも多く含まれる。したがって、提案の内容を吟味する作業がプロジェクトオーナーの負担となっている。

プロジェクトオーナーの負担を軽減するために、提案の良し悪しを自動で判定し、有益提案を抽出する手法が研究されている [2]。本稿では、有益提案を抽出する手法をシステムとして実現することを検討し、有益提案の抽出結果の一例を示す。

2. 提案の自動判別手法

提案の自動判別手法 [2] について述べる。本手法は、すでに議論の終了した過去の提案にまつわるユーザの行動を学習し、新しく発案された提案が有益提案である確率を算出する。プロジェクトオーナーは確率の高い順に提案を対応することで、効率的に有益提案に対応できる。

提案がプロジェクトにマージされたか否かは GitHub のデータから機械的に判別できるため、このデータを教師データとして、以下の 5 つの素性によって学習する。なお、学習のアルゴリズムとしてランダムフォレストを使用する。

- 素性 1 提案が発案されたブランチの活発度
- 素性 2 提案で変更されたファイルの活発度
- 素性 3 提案で変更されたソースコードの行数
- 素性 4 提案で変更されたファイルの数
- 素性 5 提案が含む commit の数

3. 有益提案の抽出支援システムの設計

3.1 データの収集方式

3.1.1 GHTorrent によるデータの収集

GitHub からデータを収集する既存システムの 1 つに、GHTorrent[3]がある。GHTorrent は、GitHub の全データを提供することを目的として開発、運用がされている。

2 章で述べた手法の研究では、GHTorrent からデータを取得して学習を行っていた。本稿で設計する有益

表 1 有益提案の抽出支援システムが GitHub から収集するデータ

データ名	用途
作成者 作成日時	初めて作成した提案か否かの判別に用いる
マージの有無 commit のハッシュ値 commit message 提案の議論内容	提案がプロジェクトに採用されたか否かを機械的に判別するのに用いる
提案先のブランチ名 変更されたファイル名 変更されたコード行数 変更されたファイル数 提案が含む commit 数	学習の素性に用いる

提案の抽出支援システムで GHTorrent の使用を考えると、以下が問題となる。

- (1) 本システムで用いないデータも収集の対象になる
本システムは、提案に関係するデータのみを収集できれば良い。しかし、GHTorrent は提案に関するデータ以外にも収集しているため、提案に関するデータのみを収集する場合に比べて無駄なデータが増え、管理する必要のあるデータの量が多くなってしまう。
- (2) データが更新されるのに数日かかる
本システムは、日々の開発の中で利用されるため、議論が終了した提案を、できるだけ早く学習データに反映したい。しかし、GHTorrent はデータが更新されるのに数日かかるため、提案を議論が終了してから収集可能になるまで学習できない。

よって、本システムでは、提案の収集について、GHTorrent を使用せず、システムの一部として設計する。また、システム内で収集した提案を保存しておく。

3.1.2 有益提案の抽出支援システムによるデータの収集

本システムで提案を収集する場合、以下を考慮してシステムを設計する必要がある。

- (1) 本システムが GitHub から収集するデータ
提案が持つデータの中から、本システムが収集するデータを表 1 に示す。表 1 のデータは、提案の自動判別手法 [2] に必要なデータである。
- (2) 提案を収集する規模
本システムは、1 つのプロジェクトに対して、提案の収集を行うことを想定している。

[†] 岡山大学大学院自然科学研究科, Graduate School of Natural Science and Technology, Okayama University

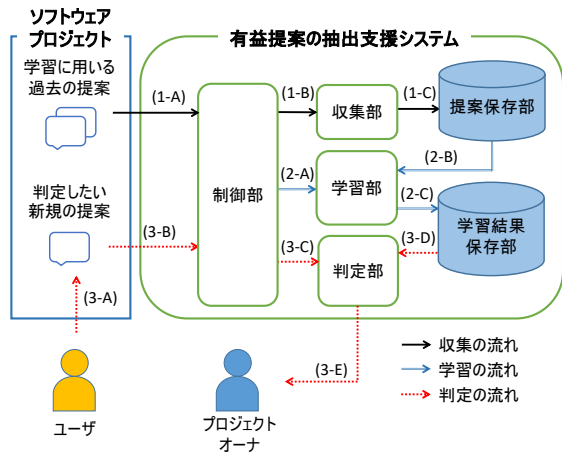


図1 有益提案の抽出支援システムの構成

Project: jquery/jquery 提案数: 14件

タイトル:

作成日時: ~ 検索

通番	タイトル	作成日時	有益である確率	操作
1	Traversing: Fix `contents()` on `<object>`s with children in IE	2019/5/7	0.646	詳細
2	Build: Test on Node 10, stop testing on Node 4 & 9	2018/4/25	0.508	詳細
⋮				
13	Ajax: Create correct URLs for jQuery.ajax() with cache: false	2017/12/22	0.431	詳細
14	[WIP] Event: Use one native handler per jQuery handler	2016/5/25	0.425	詳細

図2 有益提案の抽出結果の表示例

(3) 提案を収集する契機

提案を収集する契機としては、議論が終了した提案を、議論の終了を契機として収集する。

3.2 有益提案の抽出支援システムの構成

有益提案の抽出支援システムの構成を図1に示す。本システムは、以下の6つの部分から成る。

- 1 制御部 提案を取得し、対応する部分に渡す。また、学習に用いる過去の提案を一定数取得したら、学習部に学習の依頼を行う。
- 2 収集部 提案を提案保存部に保存する。
- 3 学習部 提案の学習を行う。
- 4 判定部 提案が有益か否か判定を行う。
- 5 提案保存部 学習に用いる過去の提案が保存される。
- 6 学習結果保存部 学習部で学習した学習済みモデルが保存される。

3.3 有益提案の抽出支援システムの処理流れ

有益提案の抽出支援システムの処理流れについて、図1に示して、以下で説明する。

- (1) 収集：制御部は、ソフトウェアプロジェクトから学習に用いる過去の提案を取得する(1-A)。制御部は、取得した提案を収集部に渡す(1-B)。収集部は、提案保存部に提案を保存する(1-C)。
- (2) 学習：制御部は、学習部に学習を依頼する(2-A)。学習部は、提案保存部から提案を取得し、提案を有益提案または無益提案に分類した後、学習を行う(2-B)。学習部は、学習済みモデルを学習結果保存部に保存する(2-C)。
- (3) 判定：ユーザーは、ソフトウェアプロジェクトに提案を提案する(3-A)。制御部は、判定したい新規の提案を取得する(3-B)。制御部は、取得した提案を判定部に渡す(3-C)。判定部は、学習結果保存部から学習済みモデルを取得する(3-D)。判定部は、学習済みモデルを用いて提案の判定を行い、判定結果をプロジェクトオーナーに渡す(3-E)。

4. 有益提案の抽出支援システムの実現

有益提案の抽出支援システムによる有益提案の抽出結果の表示例を図2に示す。本システムでは、発案された提案のタイトル、作成日時、提案が有益である確率を表示する。本システムが判定した提案は一覧として、有益提案である確率の高い順に表示される。また、提案の一覧からタイトルの検索と作成日時の絞り込みができる。詳細を選択すると、GitHubのページに遷移し、提案についての詳細が確認できる。議論が終了した提案については、一覧から削除される。

本システムを利用するプロジェクトオーナーは、表示される一覧の上から順に提案の内容を吟味することで、有益提案を優先してプロジェクトに取り込める。

5. おわりに

本稿では、ソーシャルコーディングにおける有益提案の抽出支援システムの検討について述べた。

まず、提案の自動判別手法の概要を述べた。次に、有益提案の抽出システムの設計として、データの収集方式、システムの構成、およびシステムの処理流れを述べた。最後に、有益提案の抽出支援システムの実現として、有益提案の抽出結果の表示例を示した。

残された課題として、システムの評価がある。

参考文献

- [1] GitHub, Inc.: GitHub, GitHub, Inc. (online), available from <https://github.com/> (accessed 2019-6-20).
- [2] 江見圭祐, 乃村能成, 谷口秀夫: ソーシャルコーディングにおける有益提案の抽出, 情報処理学会論文誌, Vol. 59, No. 2, pp. 486-495 (2018).
- [3] Gousios, G.: The GHTorrent Dataset and Tool Suite, *Proc. 10th Working Conference on Mining Software Repositories*, MSR '13, IEEE Press, pp. 233-236 (2013).