

進捗管理

2017/4/18 築山

研究テーマ

- 見逃しレビューを ???
- 動機
 - ▶ Investigating code review quality: Do people and participation matter?
Software Maintenance and Evolution (ICSME), 2015

やったこと, やること

- サーベイ中

- やったこと

- ▶ 例の論文が参照している論文を数本読んだ
 - ソフトウェアの品質の測り方などが少しわかった
 - SZZアルゴリズム（不具合データとコミットデータを紐づける方法）少しわかった

- やること

- ▶ Buggyレビューの定義を理解する
 - どのようにBuggyレビューを抽出したのか
 - bugの定義や, レビューの期間など
- ▶ その上で, なぜBuggyレビューが発生したのかを推察
- ▶ レビュー後の不具合再発についての論文をサーベイ

予定

- とりあえずサーベイ

- ▶ 今は, Buggyレビューを特定したいというモチベーションで

- ▶ 4月, 5月半ばくらい

- Buggyレビューの定義を理解 (4月中)
 - Buggyレビューを抽出 & 実際に内容を見る (5月前半)

- ▶ 5月

- Buggyレビューを理解した上でサーベイ

- ▶ 6月

- 暫定的に研究の方向性を決める
 - 大ゼミでの研究テーマ発表… (6/26)

- ▶ 7月, 8月

- 実際にデータを取得するなど研究を始める

	4月	5月	6月	7月	8月
Buggyレビュー の理解	■				
サーベイ		■			
暫定的に 研究テーマを決定			■		
研究を始める				■	■

予定（4月 - 5月）

- 具体的に研究を進める準備を始める
 - ▶ データの取得（来週くらいを希望 できなかつたら下記からBuggy Review見る）
 - 取得場所：Mercurial（コミットデータ）, Bugzilla（不具合データ）
 - 技術：Python?

	4月4週	5月1週	5月2週	5月3週	5月4週
データ取得					
Buggy Review 目視					
テーマ再考					
手法を考える (準備)					

- 心が折れたときは, 去年取得したREOPENしたバグ表を読む
 - ▶ コミットデータは山谷さんが取得していた, Mozillaのデータから（今はない）
 - ▶ REOPENされた不具合を取得した
 - ▶ 不具合データはWeb（Bugzilla）
 - まずここを取得する方が現実的？

現段階での構想（妄想）

- ◎ 人がレビューで見逃しやすい不具合を特徴付けて特定したい
 - ▶ 無理 or もうあるかも
 - ▶ そもそも最初から埋め込まれやすい不具合を見るべき？
 - レビュー（すごい人？そのコンポーネントでの権限者）でも見逃す不具合を見たい？
 - ▶ 特定の ソフトウェア or ハードウェア 環境での不具合などかも
 - その場合は、どれくらいの期間で不具合が起こるのかわかったらいい？
 - ▶ レビューに負荷がかかりすぎていてちゃんとレビューできてなかったのかも
 - 人によって見逃す（ありそう）、どれくらいの負荷がかかっていたのかがわかったら？
 - ▶ etc...
- ◎ Buggyレビューの定義がわかるまではなんとも言えない