

# ユーザレビューから利用時の品質問題の抽出方法の提案と評価

M2017SE013 和田 敦皓

指導教員 青山 幹雄

## 1 はじめに

ユーザレビューを分析することで、公開されたアプリケーションに対するユーザの要求を得ることができる[8]。そのため、ユーザレビューから要求を抽出する方法が提案されている[3, 9]。しかし、機能要求と非機能要求を区別して抽出する方法は確立されておらず、幅広い性質を持つ非機能要求[5]の抽出にはその対象となる箇所の特定制と要求の特性による分類が必要となる。

本研究では、ユーザレビューからアプリケーション全体またはユースケースの利用時の品質に言及した非機能要求を抽出することを目的とする。そのために、ユーザレビューのレビュー対象の定義および特定制と利用時の品質への言及の有無に着目してユーザレビューを分類する方法を提案する。

## 2 研究課題

本研究では、以下の2点を研究課題とし、ユーザレビューからアプリケーションの利用時の品質に関する問題点の抽出方法(以降、利用時の品質問題)を提案する。

- (1) ユーザレビューの対象の定義
- (2) 利用時の品質に基づいた非機能要求の抽出

## 3 関連研究

### 3.1 利用時の品質欠如から発生する問題の特定制

利用時の品質による問題は、ユーザニーズに対する非機能要求のミスマッチが原因の一つと考えられている[7]。これらを見出すために、利用状況(目標、背景情報、ユーザ/システムの挙動など)の明確化と、ユースケースに対しどのような非機能要求を持ち、どの程度の性能値が必要かを特定するために要求事項の分析が有効であると示している。

### 3.2 ユーザレビューのソフトウェア開発への利用背景調査

開発者はエンドユーザからのユーザレビューを有益な情報源と認識している[8]。要求としては、ユーザ視点からユースケースが欠落しているという報告を含む機能要求と、アプリケーションのユースケースの欠陥やユーザの不満点を見出して修正することで品質を向上させるための非機能要求が有用であるとされる。

### 3.3 ユーザレビューの分析方法

ユーザレビューを収集し、開発者にとって有益な情報を抽出するための方法が提案されている。

#### (1) 有益な情報の抽出

機能要求など、開発者にとって有益な情報を抽出する方法が提案されている[3]。分類機を用いた非有益な情報の削除を行った後、ユーザレビューに要求の対象ごとのグループ分類を行う。グループ同士、グループ内のユーザレビュー同士の相対評価を行うことで、ユーザレ

ビューごとの有用性を定量化する。

#### (2) 要求の抽出

ユーザレビューを用いたアプリケーションの開発支援の方法が提案されている[9]。ユーザレビューから機能要求および非機能要求を一括に抽出するために、ユーザレビューの集合に対しトピックモデリングを行い、トピックごとに要求事項として抽出する。また、トピックの適合率の高いユーザレビューを代表例として抽出する。

## 4 アプローチ

図1に本研究のアプローチを示す。ユーザレビューの情報を非機能要求として扱う場合、アプリケーションのユースケースに対してかアプリケーション自体への言及かを判断する必要がある。また、非機能要求は対象の質に対する言及を含むため、ユーザによって発見される問題の性質ごとに分類する必要がある。そのため、アプリケーションへの言及の対象と利用時の品質[4]に基づいた非機能要求の性質を着眼点とし、2段階にユーザレビューを分類する。これにより性質の似通った非機能要求を持つユーザレビューごとの抽出を可能にする。

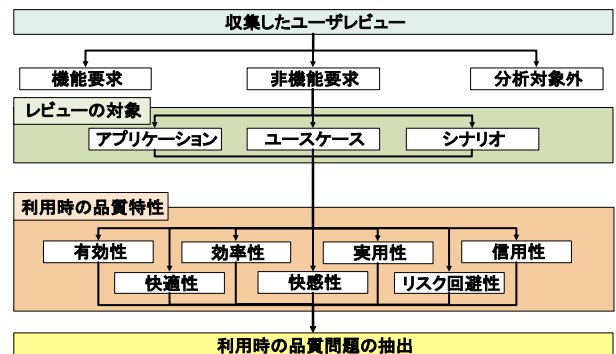


図1 アプローチ

## 5 提案方法

図2に、ユーザレビューの分類プロセスを示す。情報量を絞りこむために1文ごとに分割されたユーザレビュー(以降、レビュー文)に属性を2段階で付与することで、ユーザレビューを分類する。分類の結果、付与された属性ごとにシステムの問題点を利用時の品質問題として抽出することによって、ユーザが抱く問題点を利用時の品質ごとに抽出できる。

### 5.1 レビュー対象の定義

レビュー文のレビュー対象についての定義を行う。ユーザレビューは、その質によっては不具合報告であっても役に立たない[8]。そのため、情報抽出の段階で利用価値の高い情報を個別に抽出する必要があると考えられる。利用状況を含んでいるか、アプリケーションそのものまたはアプリケーションのユースケースに沿ったテキストになっているか

といったレビューの対象ごとの分類を行う。本研究では、レビュー対象を抽象度ごとに3種類の属性をレビュー文に付与し、レビュー文の分類を行う。

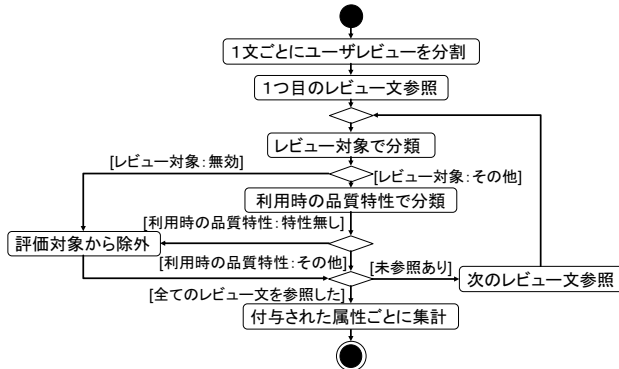


図 2 ユーザレビューの分類プロセス

### (1)アプリケーションレベル

アプリケーション自体への要求を持つレビュー文。該当アプリケーションに対するレビュー文であることが特定できることを前提とする。後述のシナリオおよびユースケースには言及されていない。

アプリケーションに対する感想および評価に関する情報の抽出と抽象的な表現を持った非機能要求の区別を目的とする。

### (2)ユースケースレベル

ユースケース[6]への言及を持つレビュー文。アプリケーションの特定ができることに加え、アプリケーションに想定されたユースケースに対する情報を含むことを前提とする。そのため、機能要求は想定されていないユースケースに対する要求として扱うため、ここには含まれない。シナリオには言及されていない。

ユーザが不満点または問題点を抱くユースケースの特定および、ユースケースに発生する問題の情報の抽出を目的とする。

### (3)シナリオレベル

シナリオ[2]への言及を持つレビュー文。ここでは、アプリケーションの特定、ユースケースに対する情報に加え、利用状況に対する情報を含む。ユースケースに対する情報に加えて、ユーザが行う行動やシステムの振る舞いの状況、またはユーザの目的が含まれている場合にシナリオレベルとして扱う。

ユースケースに発生する問題およびその利用状況の分析に利用できるデータの抽出を目的とする。

### (4)無効

レビュー対象が特定できないレビュー文。以降のプロセスでは評価の対象からは除外する。以下のような内容が含まれるレビュー文は無効として分類される。

#### (a) 機能要求

本研究では機能要求の抽出は対象としていないため、機能要求を含むが、非機能要求に対する情報を持たないレビュー文は無効として扱う。

#### (b) ミスユースケース[1]

ユーザレビューには、ユースケースまたはアプリケーションの利用そのものを阻害する要素に対する言及がされる場合が存在した。想定されたユースケース自体に発生している問題点の抽出を行うため、ミスユースケースのみの記述は無効

として扱い、ユースケースが特定できるレビュー文はユースケースのミスユースケースを抽出するためにシナリオレベルとして本研究の分析対象として扱う。

## 5.2 利用時の品質による分類

2段階目の分類では、非機能要求の特性について着目した分類を行う。ユーザレビューにはアプリケーションの利用者であるユーザが、システム利用時に抱いた不満点についての情報が含まれる。そのため、ユーザレビューに含まれる要求を非機能要求として情報を抽出する場合、利用時の品質に基づいた分類方法が有効であると考えられる。利用時の品質モデルに基づいて、レビュー文に利用時の品質特性の属性を付与する。表1に、利用時の品質特性および属性付けされるレビューの定義を示す。

表 1 利用時の品質特性

利用時の品質特性	レビュー文の定義
有効性	タスクが最後まで完了できないこと、正確に行えないことへの言及を含むレビュー文。
効率性	タスクを正確かつ完全に完了するまでに浪費した時間についての言及を含むレビュー文。
実用性	利用の結果、目標の達成状況によって得られる満足度に関する言及を含むレビュー文。
信用性	ユーザが製品またはシステムが意図した通りに動作するという確信の度合いを含むレビュー文。
快感性	個人的なニーズを満たし、ユーザが感じた満足度に関する言及を含むレビュー文。
快適性	利用時の快適さの満足度に関する言及を含むレビュー文。
リスク回避性	製品又はシステム利用の際、経済状況、人間の生活又は環境に対する潜在的なリスクに関わる言及を持つレビュー文。
特性なし	上記の品質特性を含まないレビュー文。

満足性の定義は幅広く、ユーザのニーズを満たすための要素は多方面からの分析が必要となる。そのため、副特性ごとに分析する必要があると考えられるためである。

特性無しについては、レビュー対象の特定のプロセスでアプリケーション/ユースケース/シナリオレベルのいずれかに分類されたものの、利用時の品質についての言及が存在しなかったレビュー文が分類される。

いずれの利用時の品質特性についても、ユーザが高評価を意味するレビューを行う場合が考えられるため、全てが利用時の品質問題とはならない。

## 6 SNS アプリケーションへの適用

3つのSNSのユーザレビューに提案プロセスを適用し、評価した。本研究では、Twitter、LINE、Facebookに寄せられた各100件のユーザレビューを対象に提案プロセスを適用した。表2にユーザレビューの分割結果を、表3にレビュー文の対象による分類結果の集計を示す。各アプリケーション共に、各レビュー対象への言及が抽出されたが、Twitterはシナリオが少なくアプリケーションが多いため、比較的抽象度の高いレビュー文が多いと考えられる。

表 2 ユーザレビューの分割結果

アプリケーション名	Twitter	LINE	Facebook
レビュー文総数	231	197	174

表 3 レビュー文の対象による分類結果

	アプリ ケーション	ユース ケース	シナ リオ	小計	無効	合計
Twitter	59	54	31	144	87	231
LINE	36	43	40	119	78	197
Facebook	55	33	42	130	44	174

### 6.1 Twitter のユーザレビューへの適用結果

表 4 に Twitter へのユーザレビューの分類結果を示す。アプリケーション自体およびアップデートの感想が含まれる、アプリケーションレベルの快適性に多く分類された。ユースケースレベルおよびシナリオレベルでの分類では、有効性に分類されるレビュー文が多くなった。また、ユーザが不要と感じるユースケースへの言及が多く、ユースケースレベルの実用性に多く分類された。

表 4 Twitter へのユーザレビューの分類結果

	アプリ ケーション	ユース ケース	シナリオ	合計
有効性	1	23	15	39
効率性	2	5	3	10
実用性	3	13	1	17
信用性	2	3	5	10
快感性	1	4	0	5
快適性	41	2	2	45
リスク 回避性	0	0	0	0
特性なし	8	6	4	18
合計	58	56	30	144

### 6.2 LINE のユーザレビューへの適用結果

表 5 に LINE へのユーザレビューの分類結果を示す。ユースケースの使用にあたってユーザが不満を抱くユーザインターフェースについての言及が多かったため、ユースケースレベルの快適性に多く分類された。

表 5 LINE へのユーザレビューの分類結果

	アプリ ケーション	ユース ケース	シナリオ	合計
有効性	7	26	24	57
効率性	1	2	2	5
実用性	2	1	0	3
信用性	0	2	7	9
快感性	2	0	1	3
快適性	21	9	1	31
リスク 回避性	1	0	0	1
特性なし	2	2	6	10
合計	36	42	41	119

### 6.3 Facebook のユーザレビューへの適用結果

表 6 に Facebook へのユーザレビューの分類結果を示す。アプリケーションの有効性に多く分類され、抽象的な情報ながらアプリケーションが動作中に落ちる/停止するといった致命的な不具合の報告が抽出された。

## 7 評価

### 7.1 レビュー対象ごとの利用時の品質問題の分布

図 3 にアプリケーションレベル、図 4 にユースケースレベル、図 5 にシナリオレベルごとに抽出された利用時の品質問題の

表 6 Facebook へのユーザレビューの分類結果

	アプリ ケーション	ユース ケース	シナリオ	合計
有効性	10	24	21	55
効率性	2	0	1	3
実用性	1	1	2	4
信用性	1	4	11	16
快感性	4	1	2	7
快適性	29	1	2	32
リスク 回避性	4	0	0	4
特性なし	4	2	3	9
合計	55	33	42	130

分布を示す。棒グラフは分類されたレビュー文の総数、折れ線グラフは分類結果ごとに含まれる利用時の品質問題の割合を示す。

#### (1) アプリケーションレベル

実用性、快感性、快適性が利用時の品質問題の割合が低かった。その 3 点では、他の利用時の品質特性と比較すると利用時の品質に問題点を抱く場合が少なかった。特に快適性では、アプリケーションに対して高い評価を持つユーザが使用感をユーザレビューに記述している場合が多かった。

#### (2) ユースケースレベル

快感性でのみ利用時の品質問題を含まないレビュー文が存在した。アプリケーションの場合と比較するとレビュー文単位の分析ではユースケースレベルの分類では問題点が多く含まれた。

#### (3) シナリオレベル

利用時の品質問題を含まない判断されるレビュー文は存在しなかった。レビュー文単位のシナリオレベルの分類では問題点以外を含むことは少ないと考えられる。

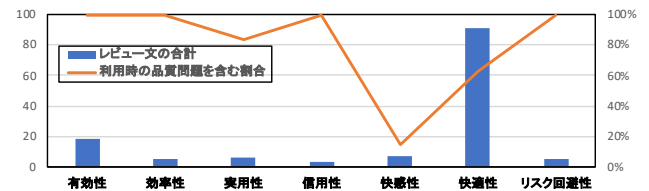


図 3 アプリケーションレベルの分布

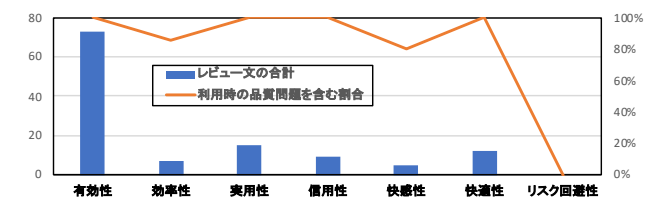


図 4 ユースケースレベルの分布

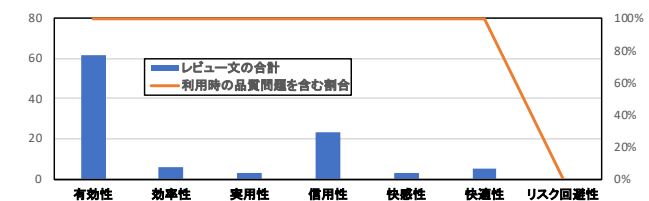


図 5 シナリオレベルの分布

### 7.2 抽出方法の評価

#### (1) ユーザレビューの対象の定義

アプリケーションレベル、ユースケースレベル、シナリオレ

ベルに分類した。その結果、ユーザレビューの抽象度ごとに分析することが可能となった。含まれる情報の定義を行なっているため、分類結果による抽象度の矛盾は発生していない。

## (2) 利用時の品質特性に基づいた非機能要求の抽出

利用時の品質を分類の属性として用いることで、ユーザ視点の非機能要求の分類を行うことが可能となった。非機能要求と判定できるレビュー文を用いて、利用時の品質に基づいた分類をおこなった。その結果、利用時の品質に問題点が含まれる箇所および問題点の特性ごとの抽出が可能となった。

## 8 考察

### 8.1 抽出された利用時の品質問題の有用性

#### (1) アプリケーションレベル

有効性、効率性、実用性、信用性では抽象的な表現の要求が多く、不具合の発生箇所の特定は困難と言える。しかし、アプリケーションが重い、動かないといった重大な問題を報告する例も存在したため、ユーザが報告する問題点の分布として考慮すべき情報となる。快感性、快適性では、ユーザがアプリケーションに求めるアップデートの方向性についての要求が多数見られ、機能追加かユーザビリティの向上かといった改善点の優先度の設定において有用と考えられる。

#### (2) ユースケースレベル

ユースケースが特定された要求であるため、ユーザが不満を抱くユースケースの特定が可能となると考えられる。しかし、ユーザの習熟度が幅広く、様々な動作環境が想定されるため、ユースケースレベルから抽出された利用時の品質問題のみでは利用時の品質を阻害する要素の特定は困難となる。

#### (3) シナリオレベル

ユーザの行動や目的といった利用状況の情報をもち、ユースケースを特定できているため、利用時の品質に問題があるユースケースの特定に加え、ユースケースに対するユーザの目的と有効性などを阻害する原因となるユーザの行動を分析することが可能となると考えられる。そのため、システム上の不具合かユーザの習熟度による問題か分析することも可能となると考えられる。

### 8.2 利用時の品質問題の抽出方法の考察

#### (1) 利用時の品質問題の定義

利用時の品質を用いてユーザレビューの分類を行うことで、非機能要求に言及するユーザレビューの特定および非機能要求の分類を行うことが可能となった。満足性に関しては、副属性を考慮した分類を行うことで、ユーザの非機能要求と実装されたアプリケーションの質との乖離を性質ごとに抽出することが可能となった。しかし、アプリケーションレベルにおいては同分類結果のレビュー文で利用する場面の異なる可能性の高い例が挙げられたため、さらなる細分化も必要となる。

#### (2) 利用時の品質問題の抽出方法

ユーザレビューの対象の抽象度を非機能要求として分類する前段階で設定することで、新たなユースケースの要求を含む機能要求と既存のユースケースを含むまたはアプリケーションへの評価を持つ非機能要求の分類が可能となった。利用時の品質ごとにユーザレビューを抽出することでユーザがユースケースやアプリケーションに抱く問題点の抽出を行うことが可能となった。

## 9 今後の課題

### 9.1 抽出された情報の優先度

抽出された利用時の品質問題の属性ごとに、開発者にとっての有用性がある場面および利用価値が異なる。また、同属性として分類された利用時の品質問題の中にも利用できる場面が異なる例が発見されたため、さらなる細分化を行なった分類によって保守プロセスにおける有用性を向上させる必要がある。

### 9.2 抽出されるユースケースの情報の利用方法

シナリオレベルで抽出される情報には、ユースケースに対するユーザの利用状況に関する情報が得られた。この情報に基づいて不具合が発生する状況を特定する方法を確立する必要がある。

### 9.3 自動化への考慮

ユーザレビューの分析はツールによる自動化が望ましい。利用時の品質問題の自動抽出のためには、Chen[3]らの研究のように分類を自動で行うための方法も必要となる。

## 10 まとめ

本研究では、ユーザレビューから利用時の品質問題の抽出方法について提案した。アプリケーションごとに寄せられたユーザレビューに対し、文ごとの分割と2段階の分類によってレビュー文を情報ごとに分け、性質ごとに抽出する方法を提案した。その際、レビュー文の抽象度に着目し定義したレビュー対象の分類と、非機能要求の性質に着目するための利用時の品質に基づいた分類を提案した。3つのSNSアプリケーション、Twitter、Facebook、LINE に対して本研究の提案方法を提案し、有用性を評価した。

アプリケーション開発者のレビュー抽出プロセスにおいて、非機能要求を性質ごとに抽出し、用途ごとに分類することでレビュー抽出プロセス後の要求分析が容易になることが期待される。

## 参考文献

- [1] I. Alexander, Misuse Cases: Use Cases with Hostile Intent, IEEE Software, Vol. 20, No. 1, Jan.-Feb. 2003, pp. 58-66.
- [2] J.M.Carroll, Making Use: Scenario-Based Design of Human-Computer Interactions, MIT Press, 2000 [郷 健太郎(訳), シナリオに基づく設計, 共立出版, 2003].
- [3] N. Chen, et al., AR-Miner: Mining Informative Reviews for Developers from Mobile App Marketplace, Proc. of ICSE 2014, ACM, May-Jun. 2014, pp. 767-778.
- [4] ISO/IEC 25010:2011, Software Engineering – Software Product Quality Requirements and Evaluation (SQuaRE) – System and Software Quality Models, 2011.
- [5] A. V. Lamsweerde, Requirements Engineering, Wiley, 2009.
- [6] R. Miles, et al., Learning UML 2.0, O'Reilly Media, Inc., 2006 [原隆文(訳), 入門UML2.0, オライリー・ジャパン, 2007].
- [7] 尾崎 慎哉 ほか, ユーザ視点から機能要求・非機能要求を導出するアプローチに関する検討, ヒューマンインタフェースシンポジウム 2011, ヒューマンインタフェースシンポジウム論文集(CD-ROM), Sep.2011, ROMBUNNO.1145L.
- [8] D. Pagano, et al., User Involvement in Software Evolution Practice: A Case Study, Proc. of ICSE 2013, IEEE, May 2013, pp. 953-932.
- [9] 清 雄一 ほか, レビューサイトの情報を利用したスマートフォンアプリケーションの開発支援, 第 186 回ソフトウェア工学研究会, Vol.2014-SE-186 No.4, Nov. 2014, pp. 1-8.