Vol.2017-SE-195 No.3
2017/3/12

モバイルアプリレビューの評価項目別自動スコアリングに関する研究

平野 智久[1,a)]　高田 眞吾[1,b)]

概要：近年モバイルアプリケーション (以下アプリ) 市場は大規模な拡大を続けており，同じような機能を持つアプリが多数存在している．そのためユーザはアプリの選択の際に困ることがある．ユーザがアプリを選ぶ際，ユーザレビューを参考にすることが多いが，レビューの数が膨大である場合，どのレビューを参考にすべきかの判断が難しい．本研究では，アプリ選択支援のために，Google Play から取得したアプリのユーザレビューを評価項目別に自動でスコアリングする手法を提案する．また，評価実験の結果も示す．

キーワード：モバイルアプリケーション，Google Play，ユーザレビュー，スコアリング

1. はじめに

近年スマートフォンの普及に伴いモバイルアプリケーション (以下アプリ) 市場は大規模な拡大を続けており，同じような機能を持つアプリが多数存在している．ユーザはどのアプリをダウンロードするかを決定する際に，アプリストアに掲載されている説明文，レビューを見る．しかしレビューにおけるレーティングは総合評価の点数であり，ユーザが重要視している部分とは限らない．また，説明文を全て読む行為は時間がかかり，敬遠されることが多い．このような背景から，ユーザは類似アプリのうちどれをインストールするべきか，アプリストアに掲載されている情報のみで判断することは困難になりつつある．そのため，より多角的で簡潔な評価が必要と考えられる．

ユーザがアプリを選ぶ際に，ユーザレビューを参考にするケースが多く見られる．これは第三者目線の貴重な情報になり得るが，有名なアプリであるほどレビューの数は膨大であり，どのレビューを参考にすべきかの判断は難しい．

各レビューには，アプリのレーティング（星１つ〜5 つ），タイトル，レビューの本文からなる．

多くのアプリストアにおいてアプリの総合評価は，リリース時のバージョンから最新バージョンまで，全てのアプリのバージョンへのレビューのレーティングの平均を取ったものである．この形式を用いると，現在のアプリのバージョンに対する評価が反映されにくいという問題がある．そのため，アプリストアは現在のバージョンのみの評価を掲載すべきという意見が挙がっている [1]．

本研究では，Google Play から対象アプリの最新ユーザレビューを取得し，評価項目別にスコアリングを行うことにより，ユーザのアプリ選択を支援するツールを提案する．

[1] 慶應義塾大学
Keio University
[a)] tomohisa3156@gmail.com
[b)] michigan@ics.keio.ac.jp

2. 関連研究

本節では，関連研究としてユーザレビューを開発者支援に用いる研究と，商品レビューのスコアリングの研究について述べる．

2.1 ユーザレビューを開発者支援に用いる研究

ユーザレビューには，アプリの改良に有益な情報が含まれている．そのため，ユーザレビューの意見を反映したアップデートは，そのアプリのストア上のレーティングを向上する傾向があると言われている [2]．このような背景から開発者視点でレビューを解析し，アプリケーションの保守に役立てる研究が行われている．

Villarroel ら [3] は，開発者に優先的に修正すべきバグや実装すべき機能を提示する CLAP というツールを提案した．このツールでは，ランダムフォレスト手法 [4] を用いることによって，「バグの報告」「新機能提案」「その他」の３グループにレビューをグループ分けする．その後，DBSCAN 手法 [5] を用いて「バグの報告」「新機能提案」のグループを更にクラスタリングする．最後に，各クラスタの優先度が高いか低いかを決定し，優先的に対応すべきユーザからの要求を開発者に提示する．

本研究では，開発者を支援するという目的ではなく，ア

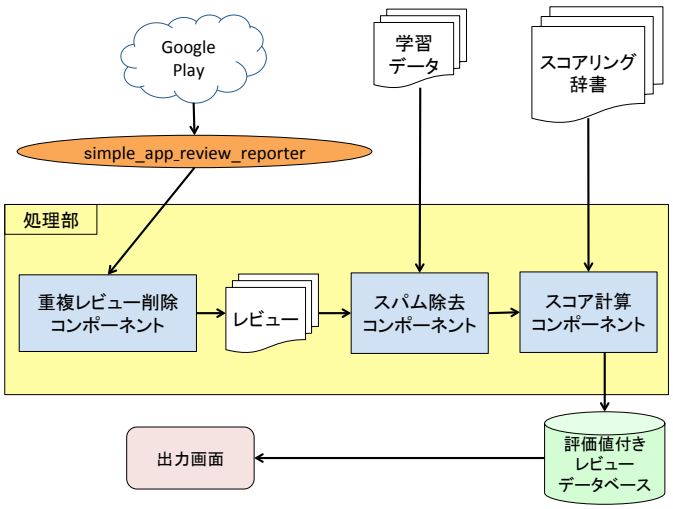

図 1 提案手法の概要

プリを利用するユーザのアプリ選択支援を目的とする．

2.2 商品レビューのスコアリングの研究

松波ら [6] はコスメアイテムに対する実際のレビューデータを用いて，スコアリングのための辞書を構築し，10 個の評価軸別の自動スコアリング実験を行った．7 段階評価で人手のスコアリングとツールのスコアリングを行った結果，平均で 0.72 の誤差があり，正解データに近いスコアリングを行えているという結果となった．

本研究では，スコアリングの対象をモバイルアプリとする．しかし，アプリのレビューはスパムレビューなどのノイズが多く含まれている．そのため，ノイズを含むレビューの除去を自動で行う必要がある．また，評価項目別のスコアリングが実際のアプリで妥当であるかを検証する必要がある．

松波らの研究では，スコアリングに利用した辞書を構築するために使ったレビューは 80 件，スコアリングの妥当性を評価する実験において評価したレビュー件数は 10 件と少なかった．本研究ではより多くのレビューを取り扱うことで精度の向上を目指す．

3. 提案

本研究では，取得したユーザレビューを評価項目別にスコアリングする手法を提案する．提案手法の流れを図 1 に示す．

まず，simple_app_review_reporter[7] を用いて Google Play からレビューを取得する．その後，重複レビュー削除コンポーネントで重複したレビューを削除する．さらに，本研究で定義するスパムレビューをスパムレビュー削除コンポーネントにて削除する．そして，スコア計算コンポーネントで評価項目別にスコアリングした結果を，評価値付きレビューデータベースに格納した後に出力画面に結果を示す．

表 1 スパムレビューの例

スパムレビューの種類	レビューの一例
アプリについて述べていないレビュー	大谷さんが辞めて、寂しくなるけど、今度の新しく野手として頑張る。
文法が正しくないレビュー	土師は徐さ婦さるさそそはすそすはそはゆはるすしふしはすす
顔文字や記号のみのレビュー	(゜ロ゜ノ)ノ
URL やメールアドレスを含むレビュー	ABC@keio.jp

3.1 重複レビューの削除コンポーネント

simple_app_review_reporter を用いたレビュー取得では，日時を指定し，レビューを 20 件ずつ取得する．そのため，前回取得してから 20 件以上投稿されていない状態で再度取得するとレビューに重複が生じる．

本研究では，このようなレビューの削除を自動で行う．

3.2 スパムレビュー削除コンポーネント

アプリストアのレビューには，ユーザがアプリを選ぶ際に参考にならないレビューが存在する．提案手法ではこのようなレビューをスパムレビューとし，スコアリングする前に自動で削除する．

本研究では以下のようなレビューをスパムレビューと定義する．スパムレビューの例を表 1 に示す．

- アプリについて述べていないレビュー
  レビュー中には，アプリについて述べていないものが存在する．例えば，ニュースアプリの場合，アプリ内で公開されている記事の感想をレビューに書き込むユーザが存在する．ユーザはこのようなレビューからアプリの情報を読み取れないため，スパムレビューと定義する．
- 文法が正しくないレビュー
  レビューの中には，単語として成立しないような文字列をただ羅列したり，連続させているレビューが存在する．このようなレビューは文章として成立しないと判断し，スパムレビューと定義する．
- 顔文字や記号のみのレビュー
  顔文字や，「！」「？」などの記号だけ書かれているレビューは，ユーザの意図が一意に読み取れないため，スパムレビューと定義する．
- URL やメールアドレスを含むレビュー
  URL やメールアドレスを含むレビューはリンク先の

情報も考慮する必要があり，レビュー単体からはユーザの意図が読み取れないため，スパムレビューと定義する．

重複を取り除いたレビューからナイーブベイズ分類器を用いて，定義したスパムレビューを取り除いた．ナイーブベイズ [8] はテキストの分類問題において，一般的に他の機械学習アルゴリズムよりも性能が良いと言われている [9]．したがって本研究の提案手法では，スパムレビューを削除する際にナイーブベイズを適用する．

本研究では，レビューの本文とタイトルを組み合わせた文章を形態素解析し，「スパムレビュー」と判断された時に，レビューをデータセットから削除する．形態素解析に用いたツールは Mecab[10] というオープンソースの形態素解析エンジンを用いる．Mecab はあらかじめ登録されている辞書に基づき形態素解析を行うが，辞書を追加することができる．本研究では，最新のオープンソースの追加辞書である mecab-ipadic-Neologd(v0.0.5)[11] を追加している．

3.3 スコア計算コンポーネント

提案手法では，著者が事前に用意したニュース閲覧アプリに関するデータセットから 8 つの評価項目を決定し，スコアリングを行う．8 つの評価項目は，データセット内のレビューの内容に関連しているものである．以下に，評価項目の詳細と，スコアリングの際に用いたスコアリング辞書を説明する．

3.3.1 評価項目

評価項目は以下の 8 つである．

- 広告：アプリ上の広告に関するレビューのスコア
- 記事：アプリ内の記事に対する内容や種類の豊富さ，タイムリー性に関するレビューのスコア
- クラッシュ：アプリが強制終了してしまう，フリーズするなど不具合に関するレビューのスコア
- デザイン：見やすさやレイアウトに関するレビューのスコア
- 評価：アプリの感想や評価に関するレビューのスコア
- 機能：アプリの機能に関するレビューのスコア
- 通信量：通信量や容量に関するレビューのスコア
- 便利さ：ユーザがアプリを利用して感じた利便性に関するレビューのスコア

表 2 に 8 項目のレビューの例を示す．

表 2 8 項目のレビューの例

評価項目	レビューの例
広告	広告多すぎ。迷惑
記事	記事の種類がたくさんあって話題に困りません。
クラッシュ	すぐに落ちる。インストールしなおしても直らない
デザイン	レイアウトがわかりやすいです
評価	毎日愛用しています!
機能	画像の拡大機能が欲しいですね。
通信量	データ使用量が増える一方です。
便利さ	サクサク進んでいい感じ

3.3.2 スコアリング辞書

本研究では松波らが提案したスコアリング手法 [6] に基づいて，スコアリングを行うための辞書を作成した．スコアリングに用いた辞書は文章に出現する単語の共起関係に基づいて構築している．構築手順は以下の通りである．

(1) レビューからアプリを評価する表現を手動で抽出しスコアを 5 段階で手動で決定する．
図 2 はキーワード共起に基づく辞書の構築方法の例を示している．例えば，「ジャンルがかなり豊富」「かなり豊富なジャンル」「ジャンルがかなり少ない」「少しジャンルが少ない」という評価表現をレビューから抽出したとする．普通が 3 点だとすると「ジャンルが豊富」という評価表現は，良い意味で捉えることができるため 4 点以上と考える．「ジャンルがかなり豊富」「かなり豊富なジャンル」はどちらも同じ意味であり，「ジャンルが豊富」という表現を「かなり」という副詞で強めている．よって二つの評価表現は 5 点と判断する．また，「ジャンルが少ない」は悪い意味で捉えることができるので 2 点以下と考える．「ジャンルがかなり少ない」は「かなり」という副詞で「ジャンルが少ない」という表現を強めているため 1 点と判断し，「少しジャンルが少ない」は「少し」という副詞で表現を弱めているため 2 点と判断する．

(2) 評価軸を表すキーワードおよび特徴と程度を示す語に分類する．
決定した評価表現を「程度」「特徴」「キーワード」の 3 つに分割する．「ジャンルがかなり豊富」「かなり豊富なジャンル」はどちらも同じ意味であるため，キーワードに「ジャンル」，キーワードの特徴を表す共起語は「豊富」，程度を表す単語は「かなり」とそれぞれ分割することで 1 つにまとめることができる．

(3) それらの共起関係と対応するスコアを決定する．
分割前に決定したスコアに基づいて，これらの共起関係をまとめた辞書にスコアを付与する．

3.3.3 スコアリングアルゴリズム

レビューに対する自動スコアリングは 8 つの項目ごとに以下の手順で行う．

(1) レビューを形態素解析し，特定項目のスコアリング辞書に登録されているキーワードの有無を確認する．

(2) キーワードを検出できた場合，同レビューに共起する特徴語及び程度を表す単語の有無を確認する．

(3) 検出できたキーワードと特徴語，程度を表す単語に基づいて，スコアリング辞書に問い合わせることで該当

図 2 キーワードの共起に基づくスコアリング辞書の構築方法

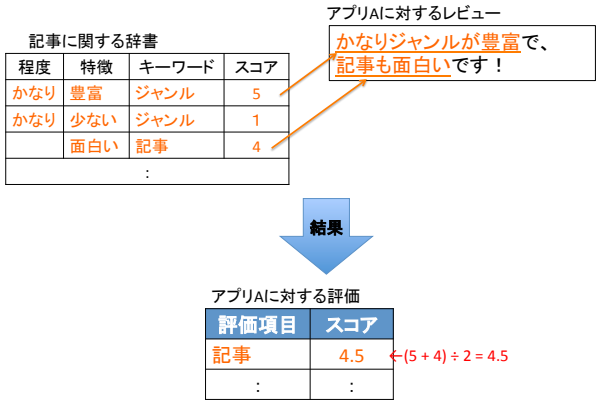

図 3 スコアリングの例

項目のスコアを取得する．

(4) 同じ項目に属するキーワードが複数検出された場合は，取得した値の平均値を該当項目のスコアに付与する．

図 3 は「かなりジャンルが豊富で、記事も面白いです！」というアプリ A に対するレビューをスコアリングしている．

まず，レビューを形態素解析すると「かなりジャンルが豊富」という表現の「ジャンル」という単語が，スコアリング辞書のキーワードに存在することが分かる．その後，共起語に特徴を表す「豊富」と程度を表す「かなり」という単語も検出でき，スコアリング辞書に問い合わせることで，記事の項目に 5 点が与えられる．同様に「記事も面白い」という表現もスコアリングされ，記事の項目に 4 点が与えられる．記事に関する表現はこれ以上見つからないため，このレビューの記事に関する項目の点数は，4.5 点 (=(5+4) ÷ 2) と決定することができる．

その後，同様の手順で他の項目もスコアリングを行い，8 つの項目の点数を決定する．

4. 評価

本節では評価実験について述べる．評価実験を行うにあたり次の Research Question(RQ) を設けた．

- RQ1：スパムレビューの削除の精度はどの程度高いのか．
  本研究で定義したスパムレビューをスパム削除コンポーネントがどの程度正確に削除できるのかを調査する．
- RQ2：提案ツールのスコアリングと手動のスコアリングはどの程度差異があるのか．
  提案手法と手動のスコアリングの精度の差異を調査する．スコアリングが行われた項目の合計と平均点数の差がツールと人手でどの程度差があるのかを比較し，レビューの本文を見ながら総合的に妥当性を調査する．
- RQ3：提案ツールはアプリ選択の支援になりうるのか．
  提案手法によってアプリ選択の支援がどの程度行えるのか調査する．被験者に対してアンケート調査を行い，スコアリング結果がどの程度アプリ選択に役立ったのかを調べた．

4.1 対象アプリ

本研究では，ニュースアプリのレビューを対象とした．利用したアプリは Smart News[12]，Yahoo![13]，Gunosy[14]，News Pass[15] の 4 つである．

ニュースアプリは，各ニュースサイトからニュースを取得し表示することができる．また，タイムリーな話題に関しては携帯の通知画面で知らせる機能があるアプリである．

ニュースアプリを選定した理由は，類似するアプリが多数あること，競合アプリの中で独占的なアプリが少ないことが挙げられる．このような背景から，ユーザがどのアプリをインストールするか迷いやすいと考えられるため，評価実験の対象アプリに設定した．

4.2 適合率，再現率，F 値

適合率 (precision)，再現率 (recall) 及び F 値 (適合率と再現率の調和平均) はしばしば統計学で用いられる評価指標である．適合率はカテゴリわけの結果の中にどの程度正解が含まれているかを示す値である．再現率は，すべてのレビューのうち，どの程度正確に分類できているのかを示す値である．F 値は適合率と再現率の調和平均である．

$$\text{適合率} = TP/(TP+FP) \quad (1)$$

$$\text{再現率} = TP/(TP+FN) \quad (2)$$

$$\text{F 値} = (2 \times \text{適合率} \times \text{再現率})/(\text{適合率}+\text{再現率}) \quad (3)$$

なお，TP，FP，FN はそれぞれ次の通りである．

- TP ＝ツールが正しく出力したカテゴリの数
- FP ＝ツールが間違って出力したカテゴリの数
- FN ＝本来正しく出力すべきカテゴリなのに，出力さ

れなかったカテゴリの数

4.3 RQ1：スパムレビューの削除の精度はどの程度高いのか．

ナイーブベイズ分類器によるスパムレビュー削除コンポーネントの精度を調査する．

4.3.1 教師データの作成

ナイーブベイズ分類器を使用するに当たって，事前に正しくスパムレビューとスパムでないレビューを教師データとして分類器に与え，学習させる必要がある．

本研究では，2016 年 10 月 15 日〜2016 年 10 月 31 日の期間で取得した各レビューを，手動でスパムレビューかスパムでないレビューかの 2 値でラベル付けを行い，教師データのデータセットを作成した．データセット内訳は全レビュー 1,159 件中，スパムレビューが 135 件，スパムでないレビューが 1024 件となった．このデータセットを用いて予備実験を行った結果，スパムレビュー分類の F 値が 0.37 と著しく低い値となった．そのため，レビューを 2016 年 12 月 31 日まで取得し続け，スパムレビューの割合を増やした．最終的にデータセットの内訳は全レビュー 1,271 件中，スパムレビューが 247 件，スパムでないレビューが 1,024 件となり，これをナイーブベイズ分類器の最終的な教師データとした．

4.3.2 テストデータの作成

テストデータは 2017 年 1 月 1 日〜2017 年 1 月 12 日の期間で取得した 528 件のレビューを全て使用した．この期間に取得したレビューを選んだ理由は，教師データ用のレビューと重複が生じることを防ぐためである．

全てのレビューにおいて手動でスパムレビューかスパムでないレビューかの 2 値でラベル付けを行い，スパムレビューが 29 件，スパムでないレビューが 499 件という内訳となった．

4.3.3 実験方法

初めに，スパム削除コンポーネントに，教師データを学習させた．その後，テストデータを入力し，2 値で分類を行い，スパムレビュー分類とスパムでないレビューの適合率，再現率，F 値をそれぞれ計算した．

4.4 RQ2：提案ツールのスコアリングと手動のスコアリングはどの程度差異があるのか．

提案ツールと人手のスコアリング精度の差異を調査する．

4.4.1 スコアリング辞書作成方法

本研究では，2016 年 10 月 15 日〜2016 年 10 月 31 日の期間で取得したレビューから，3.3.2 項で述べた方法で 8 つの評価項目それぞれのスコアリング辞書を作成した．

その後，2016 年 11 月 1 日〜2016 年 12 月 31 日に収集したレビューを用いて 2 回予備実験を行なった．人は各レビューに対して，関係すると判断した項目に対してスコアリングを行い，関係ないと判断した項目はスコアリングしなかった．そして，人がスコアリングしたのに，ツールがスコアリングを行なわなかった項目を調査した．スコアリングを行なわなかった原因がスコアリング辞書の評価表現の不足によるものであった場合は単語を追加した．また，スコアリングを行わなかった原因がスコアリング辞書の単語が正しい形態素になっていない場合は，正しい形態素に直してスコアリング辞書に追加した．

表 3 提案ツールのスパムレビュー削除の出力結果

		正解	
		スパムでない	スパム
提案ツール	スパムでない	486	13
	スパム	13	16

4.4.2 実験方法

被験者 10 名に対して，テストデータのレビューに 5 段階のスコアリングを依頼し，結果をまとめた．この時，10 名中 5 名以上がスコアリングしている項目のみ採用し，評価した被験者の平均スコアを手動のスコアリング結果とした．

その後, 提案手法を用いてテストデータと同様のレビュー 130 件のスコアリングを行ない，手動と提案ツールのスコアリング結果を比較した．

4.5 RQ3：提案手法はアプリ選びの支援になりうるのか．

提案手法がどの程度アプリ選択支援に役立っているのか実際のユーザにアンケートを行うことで調査する．

4.5.1 実験方法

本実験では，9 名の被験者にアンケートを実施した．事前に，提案ツールのスコアリング結果の閲覧を指示し，その後最も利用したいと感じるアプリとその理由を記入してもらった．

次に，実際に 4 つのアプリを並行して利用してもらい，被験者の基準で順位とその理由についてアンケートに記入してもらった．

最後に，ツールのスコアリングに関する感想を記入してもらった．

5. 結果と考察

本節では，各 RQ に対する評価実験の結果と考察について述べる．

5.1 RQ1：スパムレビューの削除の精度はどの程度高いのか．

5.1.1 結果

表 3 に提案ツールのスパムレビュー削除の出力結果を示す．スパムでないレビューは全 499 件中，486 件正確に分類することができた．一方，スパムレビューは全 29 件中，16 件正確に分類することができた．

表 4 提案ツールのスパムレビュー削除の適合率と再現率

	スパム	スパムでない
適合率	0.55	0.97
再現率	0.55	0.97
F 値	0.55	0.97

表 5 採点項目数

	全体	広告	記事	クラッシュ	デザイン	評価	機能	通信量	便利さ
手動	155	4	33	1	9	64	17	2	25
提案ツール	74	0	9	0	8	40	1	0	16
差	81	4	24	1	1	24	16	2	9

表 6 スコアリング結果

	全体	広告	記事	クラッシュ	デザイン	評価	機能	通信量	便利さ
項目数	53	0	8	0	5	28	1	0	11
手動	3.92	N/A	3.50	N/A	4.51	3.98	2.41	N/A	4.26
提案ツール	4.03	N/A	3.50	N/A	4.60	4.06	3.00	N/A	4.18
点差	0.11	N/A	0.41	N/A	0.09	0.09	0.60	N/A	-0.08

また，表 4 に，提案ツールのスパムレビュー削除の適合率と再現率を示す．スパムの適合率，再現率，F 値は全て 0.55 という結果となった．一方スパムでない適合率，再現率，F 値は全て 0.97 という結果となった．

5.1.2 考察

スパムとスパムでないレビューで分類の精度に差が出た原因に教師データの分量が挙げられる．教師データの内訳は全 1,271 件中，スパムレビューが 247 件，スパムでないレビューが 1,024 件でありスパムレビューの数が少ない．この問題は，教師データのスパムレビューの割合を増やすことで改善を見込むことができる．

一方，スパムでないレビューの分類精度は良い結果を得ることができた．表 4 の適合率を見ると 0.97 と非常に高く，実際にスコアリングされるレビューには殆どスパムレビューが含まれないと言える．要因の 1 つに教師データのスパムでないレビューの割合が高いことが挙げられる．スパムでないと判断されたテストデータのレビューに混入していたスパムレビューの例として，タイトルがどん兵衛，レビューの本文が「よろしくお願いいたします。食べたいです」がある．ナイーブベイズ分類器はグループ内で頻出する単語の有無からレビューを分類している．このレビューは「よろしくお願いいたします」がスパムでないレビューのグループに頻出している単語であるため，誤ってグループ分けされたことが考えられる．

5.2 RQ2：提案ツールのスコアリングと人手で行ったスコアリングはどの程度差異があるのか．

5.2.1 結果

表 5 は手動と提案ツールの採点項目数とその差を示す．手動は 155 項目をスコアリングしたのに対して，提案ツールは 74 項目であった．手動と提案ツールの採点項目数で最も差が小さかった項目はクラッシュとデザイン評価 (1)，最も差が大きかった項目は記事と評価（24）であった．

表 6 は項目数（表 5 の手動時と提案ツール時の採点項目数のうち，両方で現れた項目の数），各項目の平均スコア，手動と提案ツールのスコアの差を示す．全体の項目数は 53 であり，提案ツールの方が手動よりも平均 0.11 点高くスコアリングした．また，項目数が 0 でなく，提案ツールのスコアリングで最も差が小さい項目は便利さ (-0.08)，最も差が大きい項目は機能 (0.60) であった．

表 7 提案ツールの採点項目数に対する適合率，再現率，F 値

	全体	記事	デザイン	評価	機能	便利さ
適合率	0.72	0.89	0.63	0.70	1.00	0.69
再現率	0.34	0.24	0.56	0.44	0.06	0.44
F 値	0.46	0.38	0.59	0.54	0.11	0.54

また，表 7 は表 6 の項目数と表 5 の値を用いて，広告，クラッシュ，通信量を除いた 5 項目の提案ツールの採点項目数の適合率，再現率，F 値をそれぞれ示す．ここで正解データは手動のスコアリング結果を用いる．手動のスコアリングでは，被験者 10 名のうち，最低 5 名以上が同じ項目をスコアリングしている．

提案ツールが採点できた 5 項目の中で，最も高い F 値を示したのはデザイン (0.59) である．デザインの採点されたレビュー数は手動が 9 件なのに対して，提案ツールは 8 件であり，手動とツールの両方が採点したレビューは 5 件であった．

一方，最も低い F 値を示している項目は機能 (0.11) であり，次に低い項目が記事 (0.38) である．機能の採点されたレビュー数は手動が 17 件なのに対して，提案ツールは 1 件だけであった．また，手動とツールの両方が採点したレビューは 1 件であった．また，記事の採点されたレビュー数は手動が 33 件なのに対して，提案ツールは 9 件であり，手動とツールの両方が採点した 8 件であった．

全体として，適合率は 0.72 と良い結果となったが，再現率は 0.34 と課題を残す結果となった．

5.2.2 考察 1：手動と提案ツールのスコアリングの差

手動と提案ツールによるスコアリングの平均点数の差は 0.11 点と大きな差がなかった．この結果から高い精度でスコアを付与できていると言うことができる．

項目数が 0 である項目を除いて，手動と提案ツールのスコアリングで差が小さい項目は便利さ (-0.08)，デザイン (0.09)，評価 (0.09) であった．デザインが良い結果となった要因は，述べられる内容が決まっている傾向にあるため，スコアリング辞書に掲載されていない表現が少ないことが挙げられる．また，便利さと評価が良い結果となった要因は，スコアリング辞書に登録した評価表現に妥当なスコアを付与できていた事が挙げられる，

手動と提案ツールのスコアリングで差が大きい項目は機能 (0.60) と記事 (0.41) であった．機能は手動も提案ツールも採点していたレビューが 1 件のみであった．レビュー

の本文には「画像が見やすくなっていいけど検索機能がほしい」と書かれており，手動では 2.4 点，提案ツールは 3.0 点という結果となった．各被験者のスコアリング結果を見ると，このレビューに対して 2 点か 3 点を付与していたため，一概に機能の項目の点数差が大きいとは言えない．よって，機能のスコアが付与されているレビュー数を増やして調査を行う必要がある．一方，記事のスコアの差がついたレビューの中に「知識にもなるし暇潰しにちょうどよい。 使い勝手は悪くないですが、どうでもいいニュースも結構多いですね。」という文章があった．このレビューの記事の項目に手動は 2.2 点，提案ツールは 4 点を付与している．記事のスコアリング辞書にはキーワードに「ニュース」特徴語に「多い」という表現が登録されており，この表現を検出すると 4 点を付与する．しかし，「どうでもいい」という程度を表す表現は掲載されていなかったためスコアに差が開く結果となった．改善案として，スコアリング辞書の評価表現を増やす事が挙げられる．

5.2.3 考察 2：手動と提案ツールの採点項目数の差

総合の採点項目数は手動が 155 項目，提案ツールは 74 項目と半分以下となってしまった．しかし表 7 の全体の適合率は 0.72 と比較的良い値を示している．平均スコアの差も小さいことから，ユーザにスコアを提示してアプリ選択の支援を行う分には大きな問題はないと思われる．

手動で採点したが提案ツールでは採点されなかった項目について考察する．考えられる原因として，スコアリング辞書の評価表現が足りないことが挙げられる．採点されなかったレビュー数が多かった主な項目は記事や機能であった．これらの項目は固有名詞などが数多く存在していることからスコアリング辞書に加えることができなかった単語が多く存在したと思われる．このような問題の改善案として，平均スコアと同様に辞書の評価表現数を増やすことが挙げられる．

また，同じような内容でもユーザによって表現が異なることも原因の一つと考えられる．例えば「良い」と「よい」は同じ意味で使われているが，表記の仕方が違う．どちらか一方がスコリング辞書に登録されていなかった場合は，レビューが正しくスコアリングされない．また，ユーザによっては誤字に気がつかず投稿してしまう事もあり，人は考慮して採点できたとしてもツールが採点することは難しい．この問題の改善案は，レビューを形態素解析する事前処理として，同じような意味を持つ表現を統一することや，誤字脱字を修正することが挙げられる．

5.3 RQ3：提案ツールはアプリ選択の支援になりうるのか．

5.3.1 結果

被験者が提案ツールのスコアリング結果を見て最も使ってみたいと感じたアプリと，その後決定したランキングのアンケート結果を集計した結果，9 名の被験者のうち，4 名が使ってみたいアプリとランキング 1 位のアプリが一致していた．

提案ツールのスコアリングに対する感想を集計して，9 名の被験者を以下の 2 つに分けることができた．

- スコアリング結果は概ね妥当である：7 名
- スコアリング結果は妥当ではない：2 名

スコアリングの結果に概ね妥当性を感じているグループは 7 名であった．このグループのスコアリング結果に対する好意的な感想として，「ネガティブなスコアに関しては妥当性を感じる」，「News Pass と Gunosy のスコアは妥当，Yahoo!の機能以外の項目と Smart News のスコアは個人差による」があった．一方否定的な感想として，「広告量はスコアだけでは分からない．多い少ないの基準を設けてほしい」があった．また，広告や記事，デザインのスコアに関する意見は好意的な意見もあれば，否定的な意見もあった．ある被験者はこれらの項目に対して，「個人差による項目のため一概に評価できない」という意見を述べた．

スコアリング結果が妥当ではないと感じた被験者は 2 名であった．2 名とも，レビュー数と見やすさを重視してデザインの点数が高い事（それ以外の項目も相対的に高い）を理由に Smart News を使ってみたいアプリに選択していた．しかし，記事のバランスが悪い，広告が鬱陶しい，デザインが悪い，機能に満足できないと感じ Smart News のランクを，ある被験者は 3 位，もう一方は 4 位とした．両者のスコアリング結果に対する意見は，「便利さやクラッシュ，通信量は普段から使っていないと考慮できない」，「通信量は実際に測定しないと評価できない」というものであった．また，「評価項目の説明があるとアプリ選択支援によりつながるのではないか」，「複数のアプリの評価項目間での差が小さい時に，実際のレビューやアプリの画面が閲覧できると選択支援につながる」というインターフェース面での意見も得られた．

5.3.2 考察

広告や通信量は両項目ともに妥当であると意見する被験者もいたが，スコアだけでは判断できないという意見もあった．改善案として，スコアを基に多い少ないの 2 値で出力を提示する事が挙げられる．

記事やデザインの項目に関してはアプリを利用するユーザによって感じ方に個人差があるという意見が得られた．改善案として，スコア以外の情報を提供する事で，ツールを利用するユーザにアプリを使用したイメージを抱かせる事が挙げられる．例として，デザインはスコアではなくアプリのサンプル画面をユーザに提示する．記事に関してはスコア以外に標準偏差や，どのようなジャンルの記事が閲覧できるかをユーザに提示する事が挙げられる．

クラッシュ，評価，機能，便利さに関する項目は好意的な意見が多かったため，スコアリング結果の妥当性は高い

と言える．

6. 妥当性の脅威

本研究では，対象アプリのジャンルにニュース閲覧アプリを選択した．また，このジャンルのアプリのうち評価実験では Smart News，Yahoo!，Gunosy，News pass を選択している．これらは共に著者の基準で選択しているため，他のアプリを対象にした場合結果が変わる可能性がある．

本研究で用いたスコアリング辞書は，著者の基準で評価表現にスコアを付与して，8 つの評価項目毎に構築しているが，スコアを付与する基準と評価項目の選定は構築する人物によって異なる．また，スコアリング辞書の特徴を表す単語はキーワードを修飾し，程度を表す単語は特徴を表す単語を修飾する関係となるべきである．本研究では手動で辞書を作成しているため，この関係を守れていない評価表現も存在する．他にも，手動で作成することによって評価表現の誤字脱字，項目が違う辞書に追加してしまった等の誤りが考えられる．

RQ2 の評価実験では，手動では採点できているが，提案ツールは採点できていない項目が存在する事を課題にあげた．しかし，手動のスコアリングはスコアを付与する人物によって基準が異なる．また，テストデータのレビューが変わると結果も大きく変わる可能性がある．よって，実験を複数回行い，毎回スコアリングするレビューと被験者を変えるなどの改善策が必要だと考えられる．

評価実験の被験者は全員 20 代前半であった．本来ニュースの閲覧アプリは様々な年齢層から利用されているアプリであるため，被験者には様々な年齢層の人物を指定するべきである．

7. 結論

本研究では，Google Play から対象アプリの最新ユーザレビューを取得し，評価項目別にスコアリングする事で，ユーザのアプリ選択を支援する手法を提案した．評価実験の結果，スパムレビュー除去と採点項目の検出精度は改善の余地を残す結果となったが，スコアリングは手動と提案ツールで大きな点数差はないという結果となった．また，提案ツールがアプリ選択の支援につながるのかアンケートを実施したところ，「スコアリング結果が妥当な項目もある」，「アプリ選択の支援に繋がる」という意見が得られた．

今後の課題は次のようなことが挙げられる．

- 対象アプリと言語の拡大
  本研究の対象言語は日本語であったため，他の言語（特に英語）に対応できるようにするべきである．また，ニュース閲覧アプリ以外のアプリに対応することで提案ツールの一般性を示す必要があると思われる．
- スパムレビュー除去・採点項目数検出の精度向上
  スパムレビューへの分類精度の向上と採点項目の検出精度を向上させることができれば，提案手法の実用化が視野に入ると思われる．

参考文献

[1] Ruiz, I. J. M., Nagappan, M., Adams, B., Berger, T., Dienst, S. and Hassan, A. E.: Examining the Rating System Used in Mobile-App Stores, *IEEE Software*, Vol. 33, No. 6, pp. 86–92 (2016).
[2] Palomba, F., Vásquez, M. L., Bavota, G., Oliveto, R., Penta, M. D., Poshyvanyk, D. and Lucia, A. D.: User reviews matter! Tracking crowdsourced reviews to support evolution of successful apps, *2015 IEEE International Conference on Software Maintenance and Evolution, (ICSME 2015)*, pp. 291–300 (2015).
[3] Villarroel, L., Bavota, G., Russo, B., Oliveto, R. and Penta, M. D.: Release planning of mobile apps based on user reviews, *Proc. of 38th International Conference on Software Engineering, (ICSE 2016)*, pp. 14–24 (2016).
[4] Breiman, L.: Random forests, *Machine Learning*, Vol. 45, No. 1, pp. 5–32 (2001).
[5] Ester, M., Kriegel, H., Sander, J. and Xu, X.: A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise, *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96)*, pp. 226–231 (1996).
[6] 松波友稀，上田真由美，中島伸介，Oliveto, R.，階上猛，岩崎素直，O'Donovan, J.，Kang, B.：コスメアイテム評価表現辞書を用いた評価項目別レビュー自動スコアリング方式，第 9 回データ工学と情報マネジメントに関するフォーラム 2016 B1-1.
[7] *simple_app_review_reporter*, `https://github.com/KazuCocoa/simple_app_review_reporter`. [Accessed Dec. 26, 2016].
[8] Bird, S., Klein, E. and Loper, E.: *Natural language processing with Python*, O'Reilly (2009).
[9] Maalej, W. and Nabil, H.: Bug report, feature request, or simply praise? On automatically classifying app reviews, *23rd IEEE International Requirements Engineering Conference, (RE 2015)*, pp. 116–125 (2015).
[10] Kudo, T.: *MeCab:Yet Another Part-of-Speech and Morphological Analyzer*, `http://taku910.github.io/mecab/`. [Accessed Dec. 26, 2016].
[11] *MeCab IPADIC NEologd*, `https://github.com/neologd/mecab-ipadic-neologd/blob/master/README.ja.md`. [Accessed Dec. 26, 2016].
[12] *Smart News*, `https://play.google.com/store/apps/details?id=jp.gocro.smartnews.android&hl=ja`. [Accessed Jan. 23, 2017].
[13] *Yahoo!*, `https://play.google.com/store/apps/details?id=jp.co.yahoo.android.yjtop&hl=ja`. [Accessed Jan. 23, 2017].
[14] *Gunosy*, `https://play.google.com/store/apps/details?id=com.gunosy.android&hl=ja`. [Accessed Jan. 23, 2017].
[15] *News Pass*, `https://play.google.com/store/apps/details?id=com.kddi.android.newspass&hl=ja`. [Accessed Jan. 23, 2017].