

## 実利用状況を活用したモバイルアプリケーション情報共有システムの検討 Application Information Sharing System Based on Real Usage of Smartphone

関根 貴寛<sup>†</sup> 大島 浩太<sup>‡</sup> 寺田 松昭<sup>‡</sup>  
Takahiro Sekine Kohta Ohshima Matsuaki Terada

### 1. はじめに

スマートフォンの急速な普及に伴い、スマートフォンを対象としたアプリケーションマーケットが急速に成長している。Android Market[1]では 40 万のアプリケーションが公開されている(2012 年 6 月現在)。その結果、ユーザが膨大なアプリケーションの中から、自身にとって利便性の高いアプリケーションを発見するのが難しいのが現状である。既存の解決策に、アプリケーションのダウンロード履歴やレーティングを用いる手法がある。本手法はアマゾンのおすすめエンジンにも利用されている。しかし、書籍とアプリケーションは価格面での違いがある。Android Market では全体の約 70%が無料で公開されており、ユーザは多くの無料アプリケーションをダウンロードできる。また、ダウンロードしたアプリケーションをユーザが必ずしも継続的に利用しているわけではない。そのため、ダウンロード履歴によるおすすめだけでは不十分である。

近年の研究では、Twitter[2]を用いた方法や、ユーザの状況に適したアプリケーション情報を推薦する方式が提案されている[3][4]。これらのシステムは、その場で便利に利用できるアプリケーションが取得できるという利点がある。しかし、場所や時間帯に依存せず、ユーザごとに利便性の高いアプリケーションを推薦することも重要だと考える。

そこで本研究では、ユーザのアプリケーションの実利用状況に着目したアプリケーション情報共有システムを提案する。実利用状況とは、ユーザがアプリケーションを実際に利用した時間、頻度、時刻を示す。本研究の目的は、ユーザが実際に利用しているアプリケーションを調査し、ユーザが利便性の高いアプリケーションを容易に発見できるように支援することである。本システムは、実利用状況の収集、実利用状況を基にした情報共有ネットワークの構築、価値の高い情報の共有を特徴とする。

### 2. 提案システム概要

本システムは、モバイル端末で動作するクライアントアプリケーションと、実利用状況を解析して利便性の高いアプリケーションを提供するサーバにより構成される。

提案システムの概要を図 1 に示す。クライアントは主に 2 つのモジュールで構成する。実利用状況収集モジュールは、モバイル端末で常に動作してユーザが利用したアプリケーションを検知し、実利用状況を収集する。ロコミ情報提供モジュールは、ユーザが利用したアプリケーションに対するロコミ情報をサーバに提供し他のユーザと共有する。

サーバは主に 3 つのモジュールで構成する。データベース(以降 DB)は、クライアントから収集した情報と、ア

<sup>†</sup> 東京農工大学大学院工学府 Graduate School of Engineering, Tokyo University of Agriculture and Technology  
<sup>‡</sup> 東京農工大学大学院工学研究院 Faculty of Engineering, Tokyo University of Agriculture and Technology

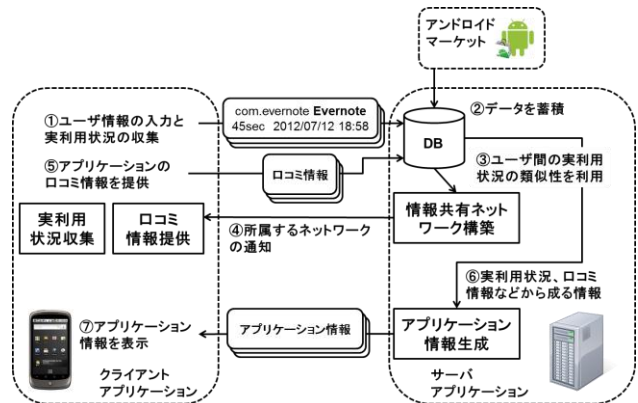


図 1 提案システム概要

アプリケーションマーケットから収集した情報を蓄積する。情報共有ネットワーク構築モジュールは、DB に蓄積された実利用状況を解析し、ユーザ間の実利用状況の類似度を求める。このようなネットワークを構築することで、嗜好の似たユーザ間で有益な情報を共有できるようになる。アプリケーション情報生成モジュールは、ネットワーク内での共有情報となるアプリケーション情報の生成を行う。アプリケーション情報とは、マーケットから取得できる情報(名前、価格、レビューなど)、利用したユーザのロコミ情報、実利用状況から成る。実利用状況を付与することで情報の信頼性を確保する。また、嗜好の似たユーザからの情報なので情報への興味・関心を確保できる。これらの情報を共有することで、各ユーザにとって利便性の高いアプリケーションの発見を支援する。以降では、本システムの 3 つの特徴について述べる。

### 3. 実利用状況収集手法

実利用状況収集モジュールは端末で常に動作する必要がある。そのため、本システムではモバイル端末として Android 端末を採用する。Android OS では、端末のバックグラウンドで動作可能な Service という機能を利用できる。また、Android API にはアプリケーションの実利用状況を取得する API は用意されていないため、実利用状況取得機能を開発した。実利用状況収集手法を図 2 に示す。

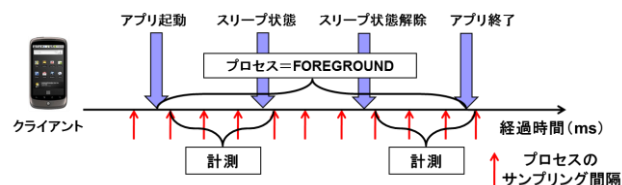


図 2 実利用状況収集手法

本手法では、Service を利用して開発モジュールをバックグラウンドで動作させ、図のように FOREGROUND プロセスを一定間隔でサンプリングする。FOREGROUND プロセスとは、端末の画面に現在表示されているアプリケーションのプロセスのことである。そのため、本システムでは、ユ

ユーザが利用しているアプリケーションを、端末の画面上に表示されているアプリケーションと定義している。

本手法のメリットは、バックグラウンドで動作するため、ユーザは意識することなく実利用状況を提供することができる点である。さらに、プロセスの検知だけでなく、端末のスリープ状態も検知することで、ユーザが本当に利用したアプリケーションを詳細に知ることができる。

デメリットは、ユーザがアプリケーションを画面上に表示したまま端末から離れた場合に、その時間も実利用状況として収集されてしまう点である。さらに、実際にアプリケーションが起動・終了した時刻と、サンプリングで得た時刻とは毎回±100ms程度の誤差が生じる。ただし、今回のシステムでは秒単位の利用時間が分かればよいので、十分な精度を確保していると考えられる。

本手法で求めた実利用状況から、ユーザのアプリケーションに対する利用度合いを示すスコアを算出する。ユーザ  $x$  のアプリケーション  $i$  に対するスコア  $S(x,i)$  は以下の計算式で求める。

$$S(x,i) = T_{x,i} * W_T + F_{x,i} * W_F \quad (1)$$

$T_{x,i}$  はアプリケーション  $i$  の利用時間、 $F_{x,i}$  は利用頻度を示す。 $W_T$  は利用時間、 $W_F$  は利用頻度に対する重みであり、アプリケーションの利用期間によって値を決める。

#### 4. 情報共有ネットワークの構築手法

本システムでは、ユーザ間の実利用状況の類似度を基に情報共有ネットワークの構築を行う。類似度の算出には、コサイン類似度を利用する。コサイン類似度とは、ベクトル空間モデルにおいて、文書同士を比較する際に用いられる計算手法である。ベクトル  $x, y$  のなす角  $\theta$  の余弦  $\cos \theta$  をコサイン類似度といい、ベクトルの向き近さを類似性の指標としたものである。1 に近ければ類似しており、0 に近ければ非類似している。以下に計算式を示す。

$$sim = \frac{x \cdot y}{|x| |y|} \quad (2)$$

$$x = \{S(x,1), S(x,2), \dots, S(x,n)\}$$

$$y = \{S(y,1), S(y,2), \dots, S(y,n)\}$$

式(2)はユーザ  $x$  とユーザ  $y$  の類似度を示しており、 $n$  はユーザ  $x, y$  に共通するアプリケーションの数である。式(2)をすべてのユーザに対して計算し、 $sim$  の値が高い  $m$  名のユーザから成るネットワークを、ユーザ  $x$  の情報共有ネットワークとする。 $m$  の値はサービス参加ユーザの人数や、 $sim$  に閾値を設定して決定する。構築された情報共有ネットワーク内で各ユーザのアプリケーション情報を共有する。

#### 5. 価値の高い情報の共有手法

本システムでは、マーケットから取得できる情報に加えて、収集した実利用状況を共有情報に加えることで情報の信頼性を確保する。さらに、嗜好の似たユーザ間で共有するので情報への興味・関心を確保する。しかし、アプリケーションが既に有名である場合や多くのユーザに利用されている場合には、共有情報の価値は低くなる。より価値の高い情報を生成するためには「情報の希少性」が必要である。つまり、各ユーザが「自分しか利用していないアプリケーション」のロコミ情報を提供することで、ネットワーク内で価値の高い情報が共有される。

そこで、本システムでは、TF-IDF を基に松本ら[5]が提案したアルゴリズム EF-ICF を応用して、情報の希少性を考慮した情報の価値判断を行う。EF-ICF とは Event Frequency-Inverse Context Frequency の略で、あるコンテキスト内で利用回数が多く、かつ他のコンテキストではあまり使われていないアプリケーションを抽出する手法である。松本らの場合、コンテキストとして位置情報と時刻を用いる。本システムでは、本手法のコンテキストの部分に「ユーザ」に置き換え、あるユーザにはよく利用され、かつ他のユーザにはあまり利用されていないアプリケーションを抽出する手法を提案する。以下にユーザ  $x$  のアプリケーション  $i$  に対する EF-ICF 値を求める式を示す。

$$eficf(x,i) = ef(x,i) * icf(i) \quad (3)$$

$$ef(x,i) = \frac{\log_2(S(x,i)+1)}{\log_2(NoE)}, \quad icf(i) = 1 + \log_2\left(\frac{NU}{Dreq(i)}\right)$$

$NoE$  はユーザ  $x$  が利用したアプリケーションの総数、 $NU$  はユーザ  $x$  が属する情報共有ネットワークのユーザ数、 $Dreq(i)$  は情報共有ネットワーク内でアプリケーション  $i$  を利用するユーザ数を示す。式(3)をすべてのユーザに対して行い、サーバ側から共有情報として価値の高いアプリケーションを各ユーザに通知する。ユーザがこれらに対するロコミ情報を提供することで、希少性のある価値の高い情報の共有が可能になると考える。アプリケーション情報を共有するまでの流れを図3に示す。

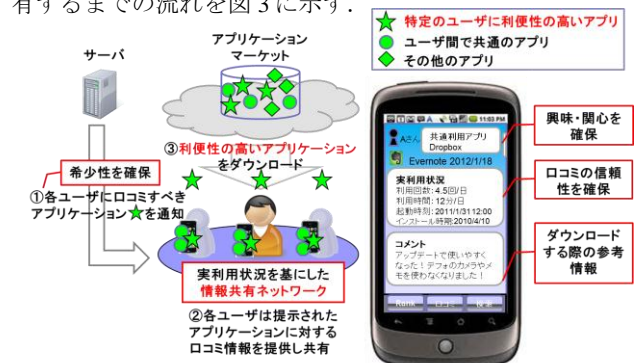


図3 アプリケーション情報共有の流れ

#### 6. まとめ

アプリケーションの利用時間、頻度、時刻を表す実利用状況を、モバイルアプリケーション情報の共有に活用したシステムの提案を行った。今後はシステムの実装および提案方式の効果検証を行う。

#### 参考文献

- [1] AppStore: <http://www.apple.com/jp/iphone/from-the-app-store/> (accessed 2012/06)
- [2] Twitter: <http://twitter.com/> (accessed 2012/06)
- [3] 矢野幹樹, 白木敦夫, 梶克彦, 松原茂樹, 河口信夫: "ユーザ生成情報を用いた携帯端末上での状況依存型サービス推薦", 「マルチメディア, 分散, 協調とモバイル(DICOMO2010)シンポジウム」, pp.221-228, (2010)
- [4] 白木敦夫, 矢野幹樹, 酒井佑太, 小澤俊介, 杉田健二, 松原茂樹, 河口信夫: "モバイルアプリケーション推薦のための Twitter 発信者の状況の推定", マルチメディア, 分散, 協調とモバイル(DICOMO2010)シンポジウム, pp.251-257, (2010)
- [5] 松本光弘, 清原良三, 沼尾正行, 栗原聡, "位置情報を含む携帯端末利用履歴からのコンテキストに基づく最適アプリケーション推定法の提案", 情報処理学会数理解モデル化と問題解決研究会(MPS) 研究報告, Vol.2010-MPS-80, No. 25, pp. 1-8, (2010)