

# モバイルアプリのファセット検索を実現する インデクシング手法

片岡 泰之<sup>1,a)</sup> 渡部 智樹<sup>1,b)</sup> 田中 清<sup>1,c)</sup> 東野 豪<sup>1,d)</sup>

**概要:** モバイル端末上で利用可能なサービスの検索支援を行うために、サービスと利用者をつなぐインタフェース ISHI(Intent of Service and Human Interface) を提案する。ISHI はサービスで提供される機能を自然言語で抽象表現で示し、サービスの意思と利用者の意思を表裏一体に兼ね備えた概念である。本研究報告では、ISHI のヒューマン・インタフェースとしての優位性を示す利用例として、モバイルアプリのファセット検索を扱う。ファセット検索の実装のために、ISHI のインデクシングの自動化手法と ISHI の重要度の測定手法を提案する。そして、モバイルアプリ特有の非構造化データを用いた本手法の効果を定量的に評価する。最後に、ISHI によるモバイルアプリのファセット検索を用いた被験者実験の結果を報告する。

## 1. はじめに

近年のモバイル機器の爆発的普及に加え、HTML5[10] による Web の進化に伴い、モバイル上で利用可能な多様なサービス（以降本論ではサービス\*<sup>1</sup>と呼ぶ）が急速に開発されている。しかしながら、その普及には様々な課題がある。情報通信白書 [11] によると、「日々増加するサービスについていけない」という情報格差問題の深刻化が報告されている。また GetJar によるモバイルアプリの調査 [1] では、利用アプリのほとんどを人気アプリが専有しておりアプリマーケットでは過度なロングテール化が進行していることが報告されている。本研究の目的は、数多あるサービスから利用者がやりたいことを実現するサービスを短時間で発見する、サービス検索支援システムの構築である。

サービス検索支援のアプローチは、利用ログ等の利用者側の統計データに基づくものと、サービス提供側のデータに基づくものに大別できる。本研究では、データの利用可能性の観点から、後者のアプローチを採用する。特に、まず「何ができるのか」を提示することで行動推薦を行った上で、次にサービス推薦をするアプローチを採用する。

しかしながら、本アプローチには以下の2つの課題がある。第1点目の課題は、各サービスに対してサービス機能

をメタデータとして付与することの自動化である。この課題1については、サービスに付属する文章、例えば web 上にあるモバイルアプリの解説文から自然言語処理によりメタデータを抽出する。第2点目の課題は、取得したメタデータの重要度の判定である。自動抽出されたメタデータにはノイズが多く、より高精度なファセット検索を実現するためには、付与するメタデータの重要度も考慮する必要がある。この課題2については、モバイルアプリ特有の構造化データである ManifestXML を利用した重要度評価手法を提案し、精度向上を図る。

本論文では、サービスの発見支援を行うために、サービスと利用者をつなぐインタフェース ISHI(Intent of Service and Human Interface) を提案する。そして、モバイルアプリのファセット検索を実現するための技術的手法を提案し、実装例と被験者実験の評価結果を報告する。

## 2. 関連研究

モバイルアプリ検索支援の関連研究を、利用者側の統計データに基づくものとサービス提供側のデータに基づくものに大別して紹介する。

### 2.1 利用者側のデータに基づくアプローチ

前者のアプローチは、アプリの検索ログやアンケートに基づき利用者の類似性や興味、利用コンテキストを分析し、アプリ検索支援を行う。ログ情報を用いることで、商品の推薦技術と同様の手法の適用が可能となり、利用者の趣味嗜好に合ったパーソナリゼされたサービス推薦が可能となる。例えば Konstas ら [2] のソーシャルグラフや Badrul

<sup>1</sup> 日本電信電話株式会社, NTT サービスエボリューション研究所, 1-1 Hikarinooka, Yokosuka, Kanagawa 239-0847, Japan

a) kataoka.yasuyuki@lab.ntt.co.jp

b) watanabe.tomoki@lab.ntt.co.jp

c) tanaka.kiyoshi@lab.ntt.co.jp

d) higashino.suguru@lab.ntt.co.jp

\*1 具体的には Android アプリ, iOS アプリ, web アプリを総称した表現とする。

ら [3] の協調フィルタリングを用いた推薦技術により、利用者の趣味嗜好に合ったパーソナリ化されたサービス推薦が可能となる。

サービスの利用ログ収集している具体例を紹介する。Grapps<sup>\*2</sup>は、3000万人の利用者のAndroidアプリの利用回数のログ情報を収集し、アプリの推薦を行っている。App Grooves<sup>\*3</sup>は、インストール済のアプリに関する質問を利用者に行い、利用者の嗜好性を判定している。大澤ら [4] はアンケート調査を行い、気分や同行者等の利用コンテキストデータを拡張し、主成分分析によりアプリが利用されるコンテキストを分析している。

しかしながら、サービス利用に関する利用者側のデータに基づくアプローチには、三点の問題があると考えられる。第一点目は、推薦精度を保つためにサービス検索ログを必要とする、コールドスタート問題である。第二点目は、利用者のプライバシー情報の取得を必要とするという問題である。第三点目は、急速に普及するアプリに対するアンケート調査には、多大な人的稼働が発生するという問題である。

## 2.2 サービス提供側のデータに基づく発見方式

後者の方式は、サービス提供側のデータを分析し、利用者が望む適切な形でサービス発見支援を行う。

Davidssonら [5] や Fengら [6] は、場所情報を用いてモバイルアプリを推薦する方式を提案した。場所情報によってアプリのインデクシングを行い、利用者の場所に依ってサービスを推薦する。本手法は、利用者側のアプリ検索操作ログを用いずに推薦を行えるというメリットがある。一方、笹島 [7] は、サービス機能(タスク)を提示することで利用者に対して行動を推薦し、次に具体的なサービスを推薦する方式を提案した。さらに、深澤 [8] がこの方式について手動で構築したシステムを用いて被験者実験を行い、検索支援に与える効果を示している。この方式には以下の2つの特徴的なメリットがある。

第一点目は行動情報はweb上の情報から抽出できるため、前者のアプローチの問題が発生しない点である。例えばブログから行動情報を抽出したり、サービスの解説文から行動情報を抽出することも可能である。第二点目は、検索行動をナビゲーションすることができる点である。従来のサービス推薦では、利用者が必要としないタイミングで不要なアプリが推薦され、利用者の満足度が低下される問題がある。原因として、推薦される対象が特定のサービスである場合、余計な機能が含まれていたりGUIの好みが合わない等、複合的な理由が考えられる。一方で、第一段階で提示された行動を気に入った利用者が、推薦された行動を実現するサービスに対して条件を簡単に加えてサービ



図 1 ISHI を用いたサービスと利用者の関連付け

スを検索できれば、従来のサービス推薦問題の一部を解決することが可能となる。

先行研究では、近年急増するモバイルアプリに対して「何が出来るのか」という行動情報を自動で付与することには至っていない。また、その行動情報によりサービスをインデクシングし、検索支援に応用するというシステムは著者らの調査では見つかっていない。

そこで本研究では、まず利用者に分かりやすい形でサービスをインデクシングするコンセプトを提案する。そして、数多あるアプリに対して行動情報なるメタデータを自動で付与する手法を検討し、さらにそれらを用いたアプリのファセット検索の実装と評価を行う。

## 3. ISHI の導入

本論文では利用者の行動情報の自然言語抽象表現である ISHI という概念を導入する。数多あるサービスに対して ISHI をメタデータとして自動的に付与することでサービスをインデクシングするコンセプトを示す。

### 3.1 ISHI の定義

本研究では、サービス機能の自然言語抽象表現を ISHI(Intent of Service and Human Interface) と定義する。サービス提供者の「この機能を提供したい」という意思と、サービス利用者の「この機能を利用したい」という意思の両者は表裏一体となっている。この意思を自然言語によって抽象表現したものが ISHI である。一般に、Intent はアプリ間やデバイス機能を連携させる概念である。例えば、Web Intents<sup>\*4</sup>や Andooid OS の Intent <sup>\*5</sup>がその代表である。これら広義の意味の Intent に対し、ISHI は、iOS, Android, Web アプリ等のモバイルアプリを含むあらゆるサービスと利用者をつなぐインタフェースとして機能する上位レイヤの概念となる。具体的には、ISHI は「写真の編

<sup>\*2</sup> Grapps, < <http://www.grapps.me/pc/pr.html> >

<sup>\*3</sup> App Grooves, <<https://itunes.apple.com/jp/app/osusumeapuri-appgrooves-ren/id423085882?mt=8>>

<sup>\*4</sup> WebIntents は W3C で 2013 年現在標準化が検討されている <[http://www.w3.org/wiki/WebIntents/Home\\_Discovery\\_and\\_Web\\_Intents](http://www.w3.org/wiki/WebIntents/Home_Discovery_and_Web_Intents)>

<sup>\*5</sup> Android APIs <<http://developer.android.com/reference/android/content/Intent.html>>

集」や「運転の診断」のように動詞と目的語で構成される自然言語抽象表現である。ISHI を用いたサービスと利用者を構造化するコンセプトを図1に示す。

### 3.2 ISHI の特徴

ISHI の特徴は、以下の四点である。第一点目は、ISHI を生成するためのデータの多様性である。ISHI はサービス機能や利用者の行動を表現しているため、利用者側だけでなくサービス側のデータから生成可能である。第二点目は、ISHI のサービス検索基盤としての利用可能性である。ISHI は利用者に対して、一目見て何が出来るかを伝え、やりたい事を実現するサービスの検索支援を行う。時には、想定しない ISHI を提示することで、利用者の気付きとなるセレンディピティを与える検索も可能となる。第三点目は、利用者の行動分析ツールとしての利用可能性である。ISHI を介してサービスを発見した場合のログ情報から、人はいつどこでどのような行動をするかという情報の分析が可能となる。第四点目は、アプリ連携基盤としての利用可能性である。利用者の行動の時系列変化を推定することで、異なるアプリを連鎖的に推薦することが可能となる。

## 4. ISHI を利用したファセット検索

ISHI のサービス検索基盤としての利用可能性の検討を行うために、モバイルアプリ、特に Android アプリを対象としたファセット検索の実装を扱う。

### 4.1 ファセット検索のシナリオ

アプリ検索を支援するには、①検索時間の短縮と②検索過程に対する満足度の向上を図る必要がある。この二点を満足させるためには、数多あるアプリから所望のアプリを発見させ、様々な条件を検索候補の絞り込みに用いるファセット検索が有効であると考えた。従来のカテゴリや人気度、料金等を用いたファセット検索に対し、本論文が提案するファセット検索のシナリオは以下の通りである。

#### (1) ISHI を一つ選択 (図2)

利用者はやりたいことをクエリとして入力する、或いは ISHI のリストからやりたいことを選択する。

#### (2) 検索アプリ候補を確認 (図3)

利用者がやりたい事を実現するアプリのリストが提示される。通常、やりたい事を実現するアプリは複数提示される。そこで次に、ISHI を追加することでアプリの絞り込みを行う。

#### (3) ISHI を検索条件に追加 (図4)

初めに入力したやりたい事に加え、さらにやりたい事(付加機能)を提示される ISHI のリストから選択する。

#### (4) 所望のサービスを選択 (図5)

追加した要求条件に合致するアプリが提示される。このアプリを選択するとインストール画面に遷移する。



図2 ISHI 選択画面



図3 サービス選択候補



図4 追加 ISHI 選択



図5 所望のサービス選択

このように ISHI を用いたファセット検索により、やりたい事(サービス機能)を2段階選択することが可能となる。

### 4.2 ISHI の機能重要性の考慮

検索条件となる ISHI には機能の重要度合いを示す指標が必要であると考えられる。例えば、多くのアプリの中で、「写真の編集」という ISHI はアプリの主機能を表現しているが、「flickr に送る」という ISHI はアプリの副機能である。このように ISHI の中でも機能表現としての重要度があると考えられる。(1)と(3)の ISHI のリストを提示する際は、より重要な ISHI を追加条件のリストの上位に配置することが望ましい。そのため、ISHI の重要度合いをメタデータとして追加付与する必要がある。この ISHI の重要度合いを、**機能重要性**と定義する。

## 5. ISHI を用いたアプリのインデクシング手法

本節では、ISHI の付与手法と、機能重要性の分析手法を述べる。本節は、①分析データ収集、②ISHI 抽出の自動化、③ISHI の機能重要性分析、④分析精度の評価、の4ステップを踏む。

### 5.1 データ収集

ISHI の分析に用いるデータとして、サービスの解説文、カテゴリ情報、そしてアクセス情報の3種のデータを注目する。アクセス情報とは、サービスがアクセスするデバイ

ス内部の機能、或いはデータを意味する。モバイルアプリの代表である Android, iOS, Web アプリの中で、これらのデータを取得可能なものは Android アプリである。<sup>\*6</sup>よって、本研究では Android アプリに限定し、アクセス情報が記述されている構造化データである ManifestXML<sup>\*7</sup>を活用する手法を検討する。

機能重要性をより正確に分析するためには、利用可能性のあるデータは積極的に活用する事が望ましい。今回対象とするデータの利用可能性については、アプリに付与された ISHI がデバイスの機能や特定の端末内データに依存している場合その ISHI は機能重要性が高い、と判断した。例えば、ManifestXML において、HARDWARE\_CONTROLS 内の CAMERA にパーミッションが与えられているアプリには、「写真の撮影」や「AR 表示」等の ISHI が多く出現する可能性が高い。これにより、機能重要性の分析が可能となると考えられる。

## 5.2 ISHI の自動抽出

サービスの解説文を自然言語処理することにより、ISHI の候補を機械的に自動抽出する。その処理結果と自動抽出精度を簡潔に示す。

### 5.2.1 ISHI の抽出処理

ISHI は、目的語と動詞から構成される単純な自然言語表現である。例えば、「経路の検索」や「経路を検索」、「経路検索」等である。これらの表現パターンを、係り受けや並列表現を考慮して web 上のアプリ解説文から機械的に抽出した。また、その際に解説文内における出現位置も取得する。さらに、「ルートを検索」と「経路の探索」などの異なる表現だが同義の ISHI を、日本語 WordNet[9] で提供されているシソーラスを適用して同一のものとしてまとめる。さらに、はてなキーワード<sup>\*8</sup>を適用し、形態素解析の精度を向上させる。

### 5.2.2 ISHI の抽出結果

Google Play<sup>\*9</sup>で提供されている Android アプリの全カテゴリから計 7532 個の日本語アプリを対象に ISHI を抽出した。その結果、合計 92633 個の ISHI が抽出された。これは、一つのアプリから平均約 12 個の ISHI を自動でメタデータとして付与できることを示している。ただし、これらの ISHI には、「運転スキルを診断」といった正確な ISHI もあれば、「情報を検索」のような抽象度が高過ぎる ISHI や「時の調整」のような意味が理解できないものも含まれ

ている。

### 5.2.3 主観評価による正解データの作成

訓練データと評価に用いる正解データを作成する。今回は、100 個のアプリの解説文から上述の処理で 1668 個の ISHI を抽出し、それぞれの ISHI に対して 10 名の試験者に以下の項目を評価してもらった。

(1) ISHI が日本語として解釈可能か

(2) 元の解説文の中で ISHI の機能重要性を 5 段階評価

この主観評価に基づくデータを正解データとし、自然言語による ISHI の自動抽出精度の評価と、機械学習による ISHI の機能重要性の評価に用いる。

### 5.2.4 ISHI の抽出精度評価

今回実装した自動抽出処理の評価を、正解データを用いて行った。計 1668 個の中で日本語として解釈が可能と判断されたものは 1313 個であった。よって各アプリに ISHI データを付与することによるアプリのインデクシングの自動化の精度は、78.7%となった。

## 5.3 機械学習による機能重要性判定

ISHI の機能重要性を、教師あり機械学習による「低中高」のマルチクラス分類問題を解くことで測定する。機能重要性が「低」と判定されたものはノイズとして処理し、自動抽出の精度向上を図る。そしてファセット検索の絞り込み条件リストを提示する際に、機能重要性が「高」の ISHI を「中」よりも優先的にすることにより、検索時間の短縮を狙う。

機械学習の手法としては、5.2.4 節で述べた正解データを訓練データとして SVM を使い、三値の ISHI の分類器を作成する。SVM の特徴ベクトルの設計を、アクセス情報、ISHI の出現位置、カテゴリ情報の処理で設計し、精度向上を図る。以降、その特徴ベクトルの詳細な設計を示す。その際に用いる数式の定義を表 1 に示す。

### 5.3.1 非自然言語処理による判定

機能重要性の評価尺度として、ISHI のアクセス情報に対する依存度を考慮した特徴量を計算する。ISHI  $I_j$  を含む全てのサービス内に共通のアクセス情報が存在すれば、ISHI  $I_j$  はデバイスの機能やデバイス内のアクセス情報に対して依存度が高い。アクセス情報と依存度が強い ISHI は重要度が高いと判断する。ISHI  $I_j$  のアクセス情報に対する依存度を示す特徴ベクトル  $f_1^j$  は、以下のように算出する。

まず、ISHI  $I_j$  を含むサービスの集合  $S^{\{j\}}$  を次のように定義する。

$$\begin{aligned} S^{\{j\}} &= \{s_i | I_j \text{ を } \mathcal{I}_i \text{ の元とするアプリ集合 } \mathcal{S} \text{ のアプリ名}\} \\ &= \{s_i | s_i \in \mathcal{S}, \mathcal{I}_i \subset \mathcal{S}, I_j \in \mathcal{I}_i\} \end{aligned} \quad (1)$$

例えば、 $I_j =$  "写真の保存" を持つサービスの集合は、 $S^{\{j\}} = \{"flickr", \dots\}$  となる。

<sup>\*6</sup> iOS からは解説文は取得できるが、アクセス情報は取得できない。また、web アプリからは Web API 等のアクセス情報と同様のデータは取得できるが、サービスの解説文を同一の形式で分析できない。

<sup>\*7</sup> Manifest XML, < <http://developer.android.com/guide/topics/manifest/manifest-intro.html> >

<sup>\*8</sup> はてなキーワード, < <http://d.hatena.ne.jp/keyword/> >

<sup>\*9</sup> Google Play, Android アプリを配信するポータルサイト < <https://play.google.com/store/> >

表 1 各変数の定義

変数	意味
$s_i$	アプリ $i$ のサービス名 (max $n$ )
$c_i$	アプリ $i$ が属するカテゴリ
$I_j$	ISHI $j$ (max $m$ )
$m_k$	アクセス情報 $k$ (max $l$ )
$C$	カテゴリの集合
$\mathcal{I}_i$	$s_i$ が持つ ISHI の集合
$\mathcal{P}_i$	$s_i$ が持つ ISHI の出現位置情報の集合
$\mathcal{M}_i$	$s_i$ のアクセス情報の集合
$\mathcal{S}_i$	$s_i$ のデータ集合 (= $\{s_i, c_i, \mathcal{M}_i, \mathcal{I}_i, \mathcal{P}_i\}$ )
$\mathcal{S}$	$\mathcal{S}_i$ を 1 つ以上含むデータ集合から成る集合
$I_{\alpha, \beta, \gamma}$	$s_\alpha$ の $\beta$ 番目の ISHI $\gamma$
$p_{\alpha, \beta, \gamma}$	$s_\alpha$ の $\beta$ 番目の $I_\gamma$ の出現位置 ( $\mathcal{P}_\alpha$ の $\beta$ 番目の元)
$m_{i, k}$	$s_i$ のアクセス情報 $k$
$\lambda$	調整パラメータ

次に, ISHI  $I_j$  を含みかつアクセス情報  $m_k$  を含むサービスの集合  $\mathcal{S}^{(j, k)}$  は, 次の様に表現できる.

$$\begin{aligned} \mathcal{S}^{(j, k)} &= \{s_i \mid \mathcal{S}^{(j)} \text{ の元のうち } m_k \text{ を持つアプリ名}\} \\ &= \{s_i \mid s_i \in \mathcal{S}^{(j)}, m_k \in \mathcal{S}_j\} \end{aligned} \quad (2)$$

最後に, ISHI  $I_j$  とアクセス情報の依存度は, ISHI が出現したサービス内に出現するアクセス情報の確率の最大値で定量化する.

$$f_1^j = \max \left\{ \frac{|\{\mathcal{S}^{(j, k)}\}|}{|\{\mathcal{S}^{(j)}\}|}, \forall k \right\} \quad (3)$$

### 5.3.2 自然言語処理による判定

機能重要性の評価尺度として, ISHI の出現位置と出現回数を考慮した特徴量を計算する. 出現位置とは, サービス解説文の先頭からの単語数に基づく距離である. サービスの機能を表現する ISHI は, サービス解説文の前半部に出現する傾向が強いという特徴を考慮し, 従来手法の tf-idf の tf を応用し, 出現位置に重みをかけてサービスの出現頻度を評価する. ある ISHI  $I_j$  の特徴ベクトル  $f_2^j$  は, 全てのサービスの中で,  $I_j$  が出現する度にその相対的な出現位置の値が小さければ重要性がより高いと判定する. 数式表現では,

$$f_2^j = \sum_i SF_i \quad (4)$$

$$SF_i = \begin{cases} \frac{1}{|\mathcal{I}_i|} \sum_{k=1}^{|\mathcal{I}_i|} \exp(-\lambda p_{i, k, \alpha}) & (if \alpha = j) \\ 0 & (otherwise) \end{cases} \quad (5)$$

で算出する. ただし,  $p_{i, k, \alpha}$  は, 表 1 の通り, ISHI  $I_j$  の出現位置を表している. 式 (4) は, 式 (5)  $SF_i$  の値を全てのサービスに対して合計した値を表している. 式 (5)  $SF_i$  とは, サービス  $i$  において ISHI  $I_j$  が出現した場合, サービ

表 2 学習精度評価

評価方法	三値分類	三値分類 ( $f_1^j$ 無)	二値分類
学習精度	0.52	0.45	0.77
予測精度	0.48	0.43	0.76

ス  $s_i$  内で出現する他の ISHI に対して出現位置を基にした相対的な重み付けを表している.

機能重要性の評価尺度として, カテゴリ内における ISHI の共起を考慮した特徴量を計算する. 機能重要性が高い ISHI は, ある ISHI  $I_j$  が属するカテゴリ  $C$  内に同じ ISHI  $I_j$  を持つサービスの数が多いと考えられる. この観点に基づいた特徴ベクトル  $f_3^j$  は, 以下のように計算する.

$$\begin{aligned} f_3^j &= \log \frac{|C|(\text{すべてのカテゴリの個数})}{|\{c_i \mid I_j \text{ を持つ全てのアプリが属するカテゴリ}\}|} \\ &= \log \frac{|C|}{|\{c_i \mid c_i \in \mathcal{S}, \mathcal{S} \ni I_j\}|} \end{aligned} \quad (6)$$

但し,  $|\{S\}|$  は集合  $S$  に含まれる元の数を表す.

## 5.4 精度評価

分類精度を評価する. 精度評価には, 適合率と再現率の調和平均である F 値を用い, 交差検定を行った値を学習精度とする. 尚, 今回用いた SVC RBF カーネルの調整パラメータ  $C$  と  $\gamma$  を, グリッド探索により選定している. さらに, 各ベクトルのスケールには, 標準偏差が 1 となる標準化を施した.

精度評価の結果を表 2 に示す. まず, アクセス情報である ManifestXML を考慮した  $f_1^j$  を採用したことにより, 約 6% の精度向上が確認された. 全体の精度向上のためには, SVM に用いる特徴ベクトルをもっと高次元にすることが効果的であると考えられる. 特に, 自然言語処理分野で得られている手法を参考にすることが有用であると考えられる.

## 6. 評価実験

本節では, 提案するモバイルアプリのファセット検索の効果を被験者実験によって検証する.

### 6.1 評価目的

本実験の目的は, 提案する ISHI を用いたファセット検索が下記の二つの項目に与える影響を評価することである.

【項目 1】 検索時間の短縮は達成されたか

【項目 2】 検索過程に対する満足度は向上したか

尚, 今回の報告では予備検証という位置付けで, 4 人の被験者を対象に実験を行った. 本実験では, 4 節で述べたシナリオに基づいたファセット検索を実装し, これを用いた被験者実験で評価を行った.

### 6.2 評価実験手法

被験者に対し, 「写真の編集をしたい」「天気予報を知

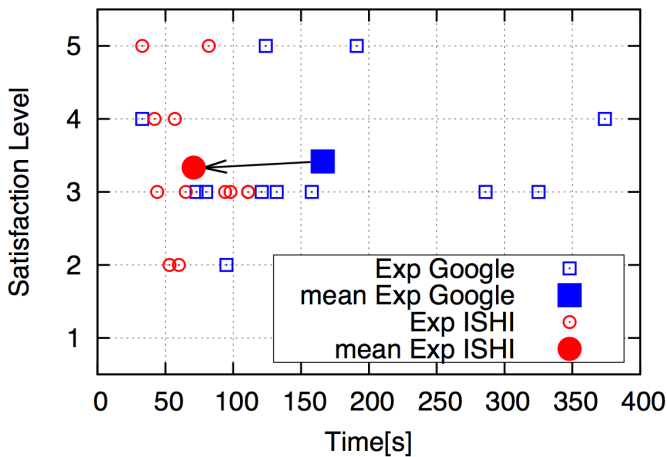


図 6 機能重要性の効果検証実験結果

りたい」「写真の共有をしたい」という3つのやりたい事を叶えるサービスを選定してもらうというタスクを与えた。このタスクを、提案手法である機能重要性を考慮したファセット検索と、Google Play上で被験者の自由な検索それぞれで実行してもらった。評価は、利用者が一つのサービスを選択するのに要した時間と、検索結果に対する満足度の主観評価で行う。主観評価の基準の目安は、選択したサービスが被験者の欲求を十分満足すると期待できる場合は5を、最低限満足すると期待できる場合は1とした。

### 6.3 実験結果

機能重要性の効果検証実験の結果を図6に示す。Google Play上での検索行動では、平均検索時間が166[s]で満足度が3.4となったのに対し、提案するISHIによるファセット検索では、平均検索時間が70.8[s]で満足度は3.3となった。

### 6.4 考察

実験結果より、ISHIを用いたアプリのファセット検索はGoogle Playでの検索方式と比較し、検索結果に対する満足度を保ちつつも、検索時間を約60%短くする効果がある。この結果から、ISHIを用いたファセット検索は、利用者のアプリ検索時における冗長な情報をうまく排除することに成功していると考えられる。検索の満足度を高めるためには、不人気なだが優良なアプリを発見するアプリ検索方式が必要であると考えられる。

今後の評価として、検索結果とアプリの人気度の関係を評価したい。従来は利用アプリのロングテールの問題があったのに対し、提案するファセット検索ではマイナーなアプリの発見に効果があるか否かを検証する。また、様々な被験者に対して実験を行い、提示されたISHIからいかに新たな気づきを得られたかを評価検証したい。

## 7. おわりに

本論文では、近年急速に普及するモバイルアプリを対象

にアプリのファセット検索を実現するインデクシング手法について提案、実装、評価を行った。まず、数多あるサービスと人の関係を構造化するISHIというIntentの上位レイヤの概念を提案した。そして、ISHIをメタデータとして自動で付与すると共に、ISHIの機能重要性を考慮することで、アプリのファセット検索の実装を行った。機能重要性の測定手法には、モバイルアプリ特有の構造化データであるManifestXMLを考慮した手法を提案した。結果として、ManifestXMLを考慮するとISHIの機能重要性の判定精度が向上することを示した。また、提案するファセット検索で被験者実験を行い、アプリ検索の満足度を損なうことなく、検索時間のみ60%短縮する効果がある、という予備評価を得た。

## 参考文献

- [1] S. Kent and A. Kamal “GetJar mobile application recommendations with very sparse datasets” In *18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp.204–212, 2012 KDD '12
- [2] I. Konstas, V. Stathopoulos and J. M Jose “On Social Networks and Collaborative Recommendation” In *32nd international ACM SIGIR conference on Research and development in information retrieval*, pp.195–202, 2009, SIGIR '09
- [3] S. Badrul, K. George, K. Joseph and R. John, “Item-based collaborative filtering recommendation algorithms” In *Proc. of the 10th international conference on World Wide Web*, pp.285–295, 2001, WWW'01
- [4] 大澤 純, 岩田 麻佑, 小牧 大治郎, 原 隆浩, 西尾 章治郎 “スマートフォンユーザのコンテキストと利用アプリケーションの関連性分析” 情報処理学会マルチメディア, 分散, 協調とモバイル DICOMO2012 シンポジウム, pp.1855–1862, 2012, DICOMO2012
- [5] C. Davidsson and S.Moritz “Utilizing implicit feedback and context to recommend mobile applications from first use” In *the 2011 Workshop on Context-awareness in Retrieval and Recommendation 6th International*, pp19–22, 2011, CaPR '11
- [6] G. Feng, E. Haihong, L. Ma and Yan Li “Design of Service Delivery Architecture for Enabling Composite Location-Based Services” In *6th International Conference Pervasive Computing and Applications*, pp324–329, 2011, ICPCA'11
- [7] 笹島 宗彦, 來村 徳信, 長沼 武史, 倉掛 正治 and 溝口 理一郎 “モバイルサービスのタスク志向型メニュー搭載を目指して-ユーザ行動モデル記述方式とその利用についての一考察-” 情報処理学会 研究報告, no.68, pp.57–64, 情報処理学会, 2007-07
- [8] 深澤 佑介, 長沼 武史, 藤井 邦浩 and 倉掛 正治 “タスク志向型モバイルサービスナビシステムの拡張とユーザ評価” 情報処理学会論文誌, vol.50, no.1, pp.159–170, 情報処理学会, 2009-01
- [9] F. Bond, T. Baldwin, R. Fothergill and K. Uchimoto, “Japanese SemCor: A Sense-tagged Corpus of Japanese”, In *The 6th International Conference of the Global WordNet Association*, GWC, 2012
- [10] W3C, W3C, <http://www.w3.org/>
- [11] “平成 23 年度 情報通信白書”, 総務省, 2011