情報処理学会論文誌 Vol.53 No.7 1783–1796 (July 2012)

閲覧Webページからの第1検索キーワード抽出に基づく検索支援

渡辺 奈夕子[1,a)] 岡本 昌之[1] 菊池 匡晃[2] 飯田 貴之[2] 佐々木 健太[1]
堀内 健介[2] 山崎 智弘[1] 大村 寿美[2] 服部 正典[1]

受付日 2011年10月24日, 採録日 2012年5月12日

概要：モバイル機器など，ハードウェアキーボードを用いないデバイスを用いたWeb検索が一般的になりつつあるが，これらの機器によるクエリ入力は手間がかかる．クエリ予測はモバイル検索にとって有効な手法と考えられるが，クエリ拡張やクエリ推薦について多くの研究が行われているのと比べると，検索クエリ中の1番目の検索キーワードに対してはほとんど注意が払われていない．本論文では，タッチ操作のような簡単な操作によりユーザが現在閲覧中のWebページと関連する情報を検索することを支援する，第1検索キーワード提示手法を提案する．提案手法は本文抽出，検索キーワード候補抽出，およびスコアリングにより構成される．また，スマートフォン上において実用的な性能で動作する，日本語Webページからのワンタッチ検索インタフェースを実装した．299 URLのWebページを用いたクローズド評価および14名のユーザによるオープン評価を通じ，提案手法が実用的な精度を達成したことを確認した．

キーワード：Web検索，クエリ予測，検索キーワード抽出，本文抽出，固有表現抽出

Search Support Based on First Query Term Extraction from Current Browsed Webpage

Nayuko Watanabe[1,a)] Masayuki Okamoto[1] Masaaki Kikuchi[2]
Takayuki Iida[2] Kenta Sasaki[1] Kensuke Horiuchi[2]
Tomohiro Yamasaki[1] Sumi Omura[2] Masanori Hattori[1]

Received: October 24, 2011, Accepted: May 12, 2012

Abstract: Inputting query terms on a device without keyboard, such as a mobile terminal, is frustrating because of device limitations though mobile web search is becoming popular. Query prediction is a promising approach for mobile search. In the literature, however, little attention has been paid to the first query term in a search query, though there are many reports on query expansion or second query term recommendation. In this paper, we propose a first query prediction method that enables users to search related information for the current browsed webpage with an easier interface such as touch operation. The proposed method consists of body-text extraction, candidate query term extraction, and a scoring process. We also implemented a one-touch search application that works for Japanese webpages at a practical speed on a smartphone. According to our closed evaluation with 299 webpages and open evaluation with 14 users, our method achieved practical quality.

Keywords: Web search, query prediction, query term extraction, body-text extraction, named entity recognition

[1] 株式会社東芝 研究開発センター
Corporate Research & Development Center, Toshiba Corporation, Kawasaki, Kanagawa 212–8582, Japan
[2] 株式会社東芝 デジタルプロダクツ&サービス社
Digital Products & Services Company, Toshiba Corporation, Ome, Tokyo 198–8710, Japan
[a)] nayuko.watanabe@toshiba.co.jp

1. はじめに

本論文の目的は，閲覧中のWebページに関連する情報の検索を支援するために，検索クエリ中の1番目の検索キーワードを自動抽出する手法の提案である．本手法はスマー

トフォンや組み込み機器のユーザ，あるいはキーボード入力に慣れていないPCアプリケーションのユーザが，検索単語を入力することなく，数回のタッチ操作だけでの関連情報検索を可能とするものである．

携帯電話・スマートフォンといったモバイル端末によるWeb検索は，PCの場合と同様一般的な操作になりつつある[12]．特に，タッチパネルを備え，ハードウェアのキーボードを内蔵しない端末で検索する機会が増えつつある．しかし，PCでのWeb検索と比べると情報アクセス機能・サービスを達成するうえでクエリ入力の手間は依然として大きな課題である．実際，携帯電話による検索活動の数はPCによる場合と比べると少ないことが報告されている[12]．

文字入力の負担を減らす試みとして，音声を発することのできる環境であれば音声検索サービス（たとえばGoogle Voice Search[*1]）のように指先の操作をともなわない検索方式も利用可能であるが，いつでも声を出して検索できる状況であるとは限らない．また，音声・文字入力いずれの場合も一定の入力ステップ数あるいは入力時間がかかるため，できるだけ少ない操作回数・操作時間で入力できる方式が望まれる．

検索支援に関する課題の1つである，クエリ入力コスト削減を目的とする研究はこれまでに数多く行われている．主要なアプローチはクエリフリー検索とクエリ推薦の2種類である．クエリフリー検索[8]は操作コストを最小にする情報検索技術である．このアプローチはユーザが様々な関連情報を必要とする場合に関連情報を提示する．クエリ推薦は，複数キーワードを利用した検索において2番目またはそれ以降の検索キーワードを提示する手法である．このアプローチはトピックに関するより特定の情報を調べるための手がかりを与えることでユーザを満足させるものである．しかし，ユーザはしばしば明確な検索意図を持っているにもかかわらず閲覧Webページの内容に基づき検索クエリ中1番目の検索キーワード（**第1検索キーワード**）を抽出する研究はほとんど行われていない．

あらゆる場面でユーザが検索する可能性のある単語を推測するのは困難であるが，Web閲覧を起点とした情報検索では，現在閲覧中のページについて関連する情報を調べる傾向が強いため，閲覧ページの関連情報検索に限定してもある程度ユーザの要求を満たせると考えられる．たとえば，ニュース関連のページを読んだ直後のWeb検索クエリはそのページについての単語であることが多い[23]．これはPC上のWeb閲覧を対象とした調査結果ではあるが，スマートフォンなどを用いたWeb閲覧においても，Webブラウザの性能向上に従いPCと同様の傾向が生じるものと考えられる．したがって，閲覧中のWebページ文章から，検索クエリの候補となる単語を抽出することで，多くの場合検索操作にかかる手間を減らすことが可能といえる．

また，検索支援におけるもう1つの課題は閲覧ページ処理および検索処理に関する通信コストとプライバシである．閲覧ページに関する処理をサーバ上で行う場合，ユーザがどのページを閲覧しているか逐一サーバに記録されることになる．クライアントでの処理負荷が下がる反面，サービス提供者に対して必要以上に詳細な情報を提供することになる．実現方法に関しても，一般的には同じURIを与えても毎回同じ内容のHTMLが取得されるとは限らないため，サーバがWebプロキシを兼ねているなどの特別な場合を除くとWebブラウザがHTMLを取得した後でHTML全体をサーバに送信する必要がある．この場合，通信量，リクエスト数，通信時間が増大する．スマートフォンなどにより携帯電話の回線を用いる場合，これらは通信料金にも影響すると考えられる．

また，検索処理に関しては，ユーザの意図と無関係にアプリケーションが自動的にWeb検索するかどうかが実装によって決まる．無線も含む通信帯域が増加しているとはいえ，ユーザが意図しない状態でアプリケーションがバックグラウンドで自動的に検索するのは通信コストがかかるうえ，検索エンジン提供者にとっても意図しないクエリが入力されることになり望ましくないと考えられる．そのため，本論文では通信コストを下げることを優先し，ユーザが意図しないバックグラウンドでの検索処理を行わない方式を用いる．

本論文では，閲覧中のWebページに関連する情報の検索を支援するための，クライアントサイドでの第1検索キーワード抽出方式を提案する．提案方式では，ユーザがWebページを閲覧するとHTMLから本文テキストを抽出し，抽出された本文テキストから検索キーワード候補を抽出しスコアを付与する．そして付与されたスコア順に検索キーワードがユーザに提示される．この方式は，特に画面の解像度が小さい端末を用いてある程度まとまった量のテキストを含むニュースサイトやブログ記事を読む場合に効果があると考えられる．たとえば，ニュース記事を読んでいるときに文中で気になる単語を見つけた場合，そのときすぐに調べるのではなく，まず記事全体を読んでから後でその単語について調べる，という利用の流れが考えられる．記事の分量が多い場合には，記事を読み終えた時点では画面には記事の一部しか表示されていない．もし，調べたい単語が記事の最初の方にある場合，画面をスクロールして目的の単語を見つけてから選択・検索操作を行う必要がある．本手法の場合は，記事を読み終えたところで検索キーワード表示操作を行うことで，スクロールしたり選択したりすることなく目的の単語で検索を行うことが可能となる．

また，ニュースサイトなどでは，Webページの中で，表示内容に関連する他のページへのハイパーリンクが提供されている場合もある．しかし，多くの場合は同じサイト内

[*1] http://www.google.com/mobile/iphone/

へのリンクであるため，他のサイトも含めて関連する Web ページを調べる場合は検索手段が必要である．

著者らは本方式をスマートフォン上に実装し，実用的な性能で動作することを確認するとともに，検索キーワード入力の負担を減らすために必要な精度が得られることを検証した．

本手法の実装に際しては，どのような単語が検索キーワードとして有用であるかを 8 つのジャンルからなる 151 の日本語 Web ページを対象とした調査を通じて決定した．また，抽出された単語がどの程度選択されたかをスマートフォン上に実装されたアプリケーションのオープン評価を通じて検証した．

本論文の構成は以下のとおりである．まず，2 章で関連研究について述べる．次に，3 章で提案するアプリケーションを紹介する．提案手法の各処理については 4 章で説明する．5 章では，どのような単語が抽出対象となるかの調査結果を報告する．6 章では，スマートフォンにおける実装例について述べる．そして，提案手法の効果について 7 章で報告する．

2. 関連研究

これまで，検索キーワードの入力コストを下げるために閲覧コンテクストを利用してキーワード抽出を行う手法が多数研究されてきた．主な方式として，クエリフリー検索とクエリ推薦の 2 種類がある．

クエリフリー検索は，適切なコンテンツを推薦するために文脈情報を推測する．このアプローチでは閲覧内容に基づきクエリを自動抽出し，検索結果がユーザに直接提示される．Letizia [17] はユーザの閲覧行動に基づき自動的にハイパーリンクを付与する．FIXIT [8] は故障時の症状に関するレポートをキーワードに対応付けることで，コピー機のマニュアルから修理情報を検索する．Query-free news search system [10] は，放送中の映像のクローズドキャプションから検索キーワードを抽出する．WebTelop [19] や映像ブックマーク検索 [21] は現在の TV 映像シーンから関連情報を検索する．閲覧 Web ページに対する関連情報検索では，補完情報を検索する WeBrowSearch [30] が提案されている．基本的に，これらの既存研究の目的は明示的な検索キーワード入力を行わないユーザに対する関連情報検索行動の支援である．しかしながら，検索を実行する前にクエリを評価するステップがなく，検索の実行回数が増える傾向がある．

クエリ推薦は，検索エンジン上のクエリログやパーソナルなクエリログ，広告クリックなどの情報を利用して 2 番目以降の検索キーワードを推定・提示するものである．アプローチとしては，ユーザのクエリログとユーザのクリック履歴からクエリ拡張用の単語を推薦する研究 [6], [9] や，ターゲット広告用の単語を抽出する研究 [29] がある．また，クエリ推薦に広告のクリックを用いる研究 [2] や，検索コンテクストの分析に意味ネットワークを用いる研究 [15], [16] がある．これらの手法は最初の検索キーワード入力を必要とするが，2 番目以降の検索キーワード入力コストを減らす効果がある．提案手法がこれらの手法と異なる点は「閲覧ページの中で 1 番目の検索キーワードとして使われそうな単語を見つける」ことにある．

閲覧 Web ページからの第 1 検索キーワード抽出に関する研究では，AQUAM [20] がある．AQUAM では，閲覧ページから意味情報を抽出し，異なる種類の固有表現の組合せとして複数の検索キーワードによる初期クエリが生成される．そしてこれらのクエリによる検索結果を分析することで検索キーワードを順序付けする．ここで，固有表現とは，固有名詞（人名，地名など）や日付，時間表現など情報抽出の基本構成要素となる単位である．検索クエリにおける 1 番目の検索キーワードを抽出することを目的とする点で本論文の提案手法と似ているが，検索キーワード選択の前にシステムがユーザの意図と無関係にバックグラウンドでの検索処理を必要とする点が異なる．

1 章で述べたとおり，バックグラウンドで検索を行う場合の検索支援における課題として，通信コストとプライバシがある．近年では通信帯域が増加しているとはいえ，アプリケーションによるユーザが意図しないバックグラウンドでの検索処理は通信コストがかかるうえ，検索エンジン提供者にとっても意図しないクエリが入力されることになり望ましくないと考えられる．また，サーバで処理する場合，ユーザがどの Web ページを閲覧しているか逐一サーバに記録されることになる．クライアントでの処理の負荷が下げられる反面，サービス提供者に対して必要以上に詳細な情報を提供することになる．

テキストからのキーワード抽出に関しては，情報抽出分野においても関連研究があげられる．情報抽出では，たとえば，電子メールの文章からの人名抽出のように，構造化されていないテキストから意味のある単語の抽出が行われる．ルールベースでのシステムとしては，FRUMP [7] や FASTUS [11] などに始まり，近年では自動的なルール改良手法 [18] や漸次的な情報抽出手法 [26] が提案されている．一方，本論文における提案手法は，人名などの単語抽出自体よりも，文書中のキーワードに対し「検索クエリとしての使われやすさ」という観点でキーワードを選別することを特徴とする．

また，画面上に表示されている単語を簡単に選択するための UI も提案されている．携帯電話向けクリック検索インタフェース [13] では，方向キーしか持たない携帯電話において，円による領域で検索語を指定することで検索を容易にする．本論文の提案手法は，主に Web ページの内容を読み終えた後，その時点で画面上に表示されていない単語も含め関連情報の検索に利用されやすい単語を抽出する

(a) Webページ閲覧　(b) キーワード・ジャンル選択　(c) 検索結果表示

図 1　検索支援アプリケーションの利用例

Fig. 1　Example use of search-support application.

点で，利用するタイミングが異なるといえる．

閲覧 Web ページを利用する以外の手法として，ユーザのおかれている状況を検索に利用する方式も考えられる [28]．たとえば，モバイル用途では，外出時の位置情報を利用した方式 [3] がある．ユーザのおかれた状況を起点とした検索方式は，たとえば「地名 レストラン」として位置情報に対応するレストランの検索を行うなど，「状況」を 1 番目のキーワードとして特定のジャンルの情報を検索することに主眼があるといえる．それに対し本手法は，「文書中のユーザが関心を持ちそうな単語」をとらえ，関連情報を検索することを目的とする．

なお，関連研究の多くは本論文における提案手法と補完的に活用可能であり，今後組み合わせることも考えられる．

3. ワンタッチ検索インタフェース

本章では，検索キーワード入力のコストを減らすことを目的とした Web 閲覧からの関連情報検索を支援する提案手法の概要と処理の流れを説明する．

図 1 は提案手法を用いたモバイル機器向けアプリケーションである「ワンタッチ検索インタフェース」[22] のスクリーンショットである．

本アプリケーションの利用例を以下に述べる．

- 本アプリケーションのユーザは Web ブラウザで Web ページを閲覧する．たとえば，図 1 (a) において閲覧中の Web ページは東芝科学館サイトの万年時計に関する記事である．検索支援機能が利用できる場合，画面上部中央の通知アイコンが図 1 (a) のようにハイライト表示される．
- ユーザが Web ページに関連する情報を知りたい場合，通知アイコンをタッチする．すると，アプリケーションによって検索に利用される可能性が高いと判断された検索キーワードの候補一覧がボタンとして表示される（図 1 (b)）．検索キーワードごとに用意されたボタンは押す領域（キーワード表記部分および検索サービス名部分）によって異なる検索サービスが割り当てられる．図 1 (b) では，通常のキーワード検索，ニュース検索，Wikipedia *2検索（本アプリケーションでは「トリビア」と表記）の 3 種類が選択可能である．なお，適当な検索キーワード候補が表示されなかった場合は，（図 1 (b)）下部に表示されているテキストボックスをタッチすることで，ユーザ自身が検索キーワードを入力することもできる．
- ユーザが，万年時計の作者である「田中久重」についてより深く知りたい場合，「田中久重」と書かれたボタンの「トリビア」部分をタッチすると，Wikipedia 検索が行われ，その後検索結果が表示される（図 1 (c)）．
- 図 1 (c) の検索結果の中で興味のある内容があれば，そのスニペットをタッチすることで対応する Web ページが表示される．

アプリケーション設計としては，2 個以上のキーワードを選択する方式も考えられるが，本研究では，なるべく少ない手間での検索がどの程度有用かを調べるため，2 単語以上を選択するステップを省き検索する方式とした．

このシナリオを実現するために，以下の処理が必要とされる．

HTML 文書からの本文抽出　広告や無関係なコメントなどを除く，記事やブログの本文などページの主な内容を示す文章の抽出

本文テキストからの検索キーワード候補抽出　検索キーワードとして利用される可能性の高い単語の推定・抽出

小型画面向けの検索キーワード候補選択　モバイル機器の小さな画面での表示を想定した，単語のスコア付けと絞り込み

*2　http://ja.wikipedia.org/

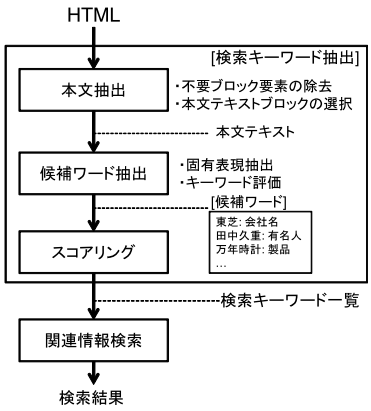

図 2 処理の流れ
Fig. 2 Process flow.

4 章において，これらの処理の実現方法について述べる．

4. 第 1 検索キーワード抽出

本章では，閲覧中の Web ページから検索クエリに用いられる可能性の高い単語を抽出する処理の流れを説明する．

本論文で扱う検索支援アプリケーションにおける処理の流れを**図 2** に示す．アプリケーションでは検索キーワード抽出と関連情報検索の 2 つのステップが実行される．そのうち，検索キーワード抽出ステップは本文抽出，候補ワード抽出，スコアリングの 3 つの処理から構成される．

4.1 本文抽出

最初に，本文抽出により閲覧中の Web ページの HTML 文書から主要な文章を抽出する．提案方式における本文抽出処理は，1 つの HTML ファイルからまとまった文章（典型的にはニュース記事やブログ記事の本文）を取り出すことを目的とする．表示上はひとまとまりとして見える文でも，HTML 記述では p，div などのタグによって区切られることがあるため，深い階層から，同じ階層に属する文を連結しながら「まとまり」を判定する方式をとる．本処理により，ニュース記事やブログのエントリの本文に相当するまとまった文章が本文テキストとして抽出され，まとまった文を含まないサイドバーや関連記事リンク，広告部分は除去される．

本文抽出処理は以下の手順により行われる．

(1) HTML のブロック要素への分割
まず，段落などを示すブロック要素タグの単位で，タグとアンカテキストを除く文字列長と句読点の個数により重みを付与する．同じ長さの文章でも，1 つの文よりは複数の文・節を含む方が情報が多いと考えられるため，文字列長だけでなく句読点の個数もスコアに用いる．一般に，文章の前半に主要な情報が記載されることが多いことから，同じ長さの文でも前半を重視するように減衰係数を設ける．具体的には，div，table，h1 など HTML のブロック要素タグで囲まれた領域（ブロック）ごとにテキストを分割し，ブロックごとの基本スコアを算出する．
HTML 中で i 番目に出現するブロックの基本スコア s_i は以下の式で算出される．

$$s_i = (l - a + pn)b_i$$
$$b_i = b_0 b_{i-1} \ (i > 0)$$

ここで，l はタグを除く内部テキスト長，a は内部テキストのうち a タグで囲まれたアンカテキストの合計長，n は句読点の個数，p は句読点の重み付け定数，b_i は HTML において後に出現するブロック要素のスコアを下げることを意図した減衰係数である．具体的には，$p = 10$，$b_0 = 0.83$ を用いた．

(2) 不要ブロックの除去
次に，文章の量が少ないブロックを除去する．アンカテキストは，リンク先のタイトルなど，閲覧中の Web ページ自体に関する文ではない場合が多いため，本文とは見なさない扱いとする．具体的には，内部テキスト長 l が閾値より短い場合，内部テキスト長に対するアンカテキスト長の比率 a/l が閾値より大きい場合，あるいはあらかじめ指定されたストップワードを含む場合は不要ブロックと見なして除去する．具体的には，l の閾値として 80 文字，アンカテキスト長比率の閾値として 0.7 を用いた．

(3) 隣接ブロックの連結
前方から順に各ブロックを評価し，隣接ブロックを連結する．連結のためのスコア c_i は以下の式により算出される．

$$c_i = s_i d_i$$
$$d_i = d_{i-1}/d_0 \ (i > 0)$$

ここで，d_i は HTML において 1 つのブロックのサイズが大きくなりすぎないよう，HTML において後に出現するブロック要素のスコアを下げることを意図した減衰係数である．c_i が閾値以上である場合には i 番目のブロックと $i-1$ 番目のブロックを連結し，閾値以下の場合は連結せず d_i を d_0 に初期化する．具体的には，$d_0 = 1.63$ を用いた．

(4) 本文として採用するブロックの選択
(3) までのステップで，文字列のまとまりが抽出される．ここから，まとまりの大きなブロックを選択して本文とする．具体的には，全ブロックをスコアの降順に並べ替え，スコア総和 Σc_i に対する前方からのスコアの合計の比率が閾値以上になるブロックまでを本文として採用する．本文として採用されたブロックを出現位置順に再度並べ替え，内部テキストをすべて連結したものを本文として抽出する．具体的には，スコア比率の閾値として 0.55 を用いた．

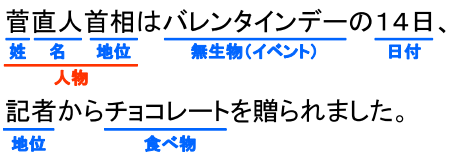

図 3 固有表現抽出の例
Fig. 3 Example of named entity recognition.

なお，上述の内容においてそれぞれのパラメータは，後述する正解データの HTML を用い予備実験を行った結果として，正解となる本文を抽出できる値を決定した．

上記手順による本文テキストのほか，HTML ヘッダの meta タグに記述される title，keywords，description に記述されるテキストがあわせて抽出される．

4.2 候補ワード抽出

次に，得られた本文テキストに対し固有表現抽出を用いることで検索キーワード候補（候補ワード）の抽出が行われる．この処理は

(1) 固有表現抽出による単語抽出
(2) 包含関係のある複数単語に対する優先度付け

という流れで実行される．以下，順を追って説明する．

まず，固有表現抽出が行われる．固有表現抽出手法には統計的な手法や文法ベースの手法が存在する [4] が，本論文では，固有表現抽出のために辞書およびルールを用いた手法を用いる．本論文の手法では，人物名，企業名，地名などの固有名詞や，日時，数量，金額などを表す文字列中の表層表現を示す単語が抽出される．また，これら単語の分類を本論文では意味クラスと呼ぶ．たとえば，「田中久重」という文字列は，「人物名」という意味クラスを持つ固有表現である．本手法は元々質問応答システム向けに開発されたもので，100 以上の意味クラスが定義されている [24], [25]．

固有表現抽出は，与えられた文に対し最初に辞書ベースの固有表現抽出を行い，その後組合せルールを用いて複合語の固有表現を抽出する．固有表現本文抽出の解析結果例を**図 3** に示す．

最初に，辞書を用いることで「菅（姓）」「直人（名）」など意味クラスが付与された固有表現が抽出される．次に，ルールを適用することで，たとえば「菅直人首相（人物）」のような複合語が固有表現として抽出される．

組合せルールの例としては，図 3 のように「姓＋名＋地位→人物」，あるいは「地名＋食品→食品」といったルールが適用されることで，それぞれ「菅（姓）＋直人（名）＋首相（地位）→菅直人首相（人物）」「神戸（地名）＋牛（食品）→神戸牛（食品）」のように抽出される．したがって，あらかじめ「菅直人首相」「神戸牛」が辞書に登録されていない場合も意味クラスが付与される．本論文で述べる実装では，8 万語の辞書および 127 種類のルールを用いている．

そして，抽出された固有表現の集合のうち，包含関係のある複数単語に対して優先度付けが行われる．

固有表現抽出の結果をそのまま用いる場合，たとえば「菅直人首相」という入力に対して「菅直人」と「菅直人首相」のように重なり合う複数の単語が抽出される場合がある．これらの単語をすべて表示する場合，表示件数が増え，ユーザにとっては操作がかえって煩雑になる．検索クエリとしては，長い文字列の方が曖昧さが少ないと考えられるが，検索結果が得られない可能性もある．この点を考慮し，包含関係にある単語が抽出された場合は以下の条件にあてはまる場合のみ短い単語を優先し，それ以外は長い単語を優先する．

- 「菅直人」と「菅直人首相」のようにフルネームと役職が連続する場合，「菅直人」を選択
- 「覚せい剤取締法違反」と「覚せい剤取締法」のように主要な単語が変わらない修飾関係の場合，「覚せい剤取締法」を選択

この結果得られた単語が，候補ワードとなる．

4.3 スコアリング

そして，前節において抽出された各候補ワードに対し，検索キーワードとしての優先順位を示すワードスコアが付与される．以下，ワードスコアを決める要素について述べる．

著者らは，以下の特徴がクエリに有用な手がかりであると仮定した．

(a) Web ページのタイトルに含まれる単語
(b) Web ページの前半に含まれる単語
(c) 頻出する単語
(d) 文字数の多い単語
(e) 補足説明（例：丸括弧で括られた文）が後に続く単語

また，以下の特徴はクエリに用いられにくい手がかりであると仮定した．

(f) 検索クエリとするには一般的すぎる単語
(g) Web サイト名自体を示す単語
(h) 記事の著者や出版社を示す単語
(i) リストで列挙された単語

これらの特徴をまとめ，以下の属性に関する各スコア（属性スコア）を各候補ワードに付与した．それぞれの数式やパラメータに関しては，スコアの増減に関する仮定をおいたうえで，後述する正解データを用いた予備実験を通じて決定した．

- s_{origin}：単語の出自（タイトル，デスクリプション，キーワード，本文）((a), (g) に対応)
  ユーザが目にするタイトル，本文に出現する場合に高いスコアを付ける．本論文では，タイトル，本文に含まれない（ユーザから見えない）デスクリプション，キーワード中の単語に対しては減点として −6.0，タ

イトル，本文中の単語に対しては 0 を付与する．

- s_{position}：本文中の出現位置（(b)，(h)，(i) に対応）および単語の文字列長（(d) に対応）本文の前の方に出現するほど，あるいは文字列がある程度の長さの場合に高いスコアを与える．具体的には，文字列長を l 文字，出現位置を p 文字目とするとき，出現位置によるスコア $s_{\mathrm{pos}}(p)$ の和（同じ単語が複数回出現する場合）と文字列長によるスコア $s_{\mathrm{len}}(l)$ を乗じたもの，つまり

$$s_{\mathrm{position}}(l,p) = s_{\mathrm{len}}(l)\sum s_{\mathrm{pos}}(p)$$

とする．$s_{\mathrm{len}}(l)$ は，l が 1 つの単語として許容できる文字列長 $L_{\min}$ までは単調増加させるが，それ以上は単語として長すぎるため文字列長 $L_{\max}$ のときに最小値 s_{lmin} をとるまで単調減少する．本論文では，下記の連続関数を用いた．

$$s_{\mathrm{len}}(l) = \begin{cases} 1 - w_{\mathrm{len}} + w_{\mathrm{len}}\left\{1 - \left(\dfrac{l}{L_{\min}} - 1\right)^2\right\} & (l \le L_{\min}) \\ 1 - w_{\mathrm{len}} + w_{\mathrm{len}}\left\{s_{\mathrm{lmin}} + \dfrac{(1 - s_{\mathrm{lmin}})(L_{\max} - l)}{L_{\max} - L_{\min}}\right\} & (L_{\min} < l \le L_{\max}) \\ 1 - w_{\mathrm{len}} + w_{\mathrm{len}} s_{\mathrm{lmin}} & (l > L_{\max}) \end{cases}$$

本論文では，$w_{\mathrm{len}} = 0.8$，$s_{\mathrm{lmin}} = 0.1$，$L_{\min} = 20$，$L_{\max} = 40$ を用いた．
また，$s_{\mathrm{pos}}(p)$ は，出現位置 p が小さい，つまり本文の最初の方にあるほど有用な単語であると考え，最大出現位置 M_{pos} において最小値 $s_{\min}$ となるまで単調減少する関数とする．本論文では，下記の連続関数を用いた．

$$s_{\mathrm{pos}}(p) = \begin{cases} s_{\min} + (1 - s_{\min})\left\{1 - 3\left(\dfrac{p}{M_{\mathrm{pos}}}\right)^2\right\} & (p \le M_{\mathrm{pos}}/3) \\ s_{\min} + 1.5(1 - s_{\min})\left(1 - \dfrac{p}{M_{\mathrm{pos}}}\right)^2 & (M_{\mathrm{pos}}/3 < p \le M_{\mathrm{pos}}) \\ s_{\min} & (p > M_{\mathrm{pos}}) \end{cases}$$

本論文では，$M_{\mathrm{pos}} = 2000$，$s_{\min} = 0.01$ を用いた．
なお，s_{len}，s_{pos} の式において線形でなく 2 乗の項を用いているのは，最大値，最小値付近の勾配を緩やかにするためである．

- s_{semfreq}：意味分類の出現頻度（(c) に対応）
列挙して表示される場合など，同じ意味クラスに分類される単語が多数出る場合，その単語が特別である可能性が低いため，スコアを下げる．本論文では，同じ意味クラスを持つ固有表現が 6 個以上出現する場合に −1 を付与する．

- s_{header}：補足説明の有無（(e) に対応）
括弧で括られた補足説明が直後に付与される単語にはスコア，本論文では 0.01 が付与される．

- s_{webidf}：WebIDF（(f) に対応）
WebIDF とは形態素による Web 検索結果件数を文書頻度（DF: document frequency）として近似した場合に計算される IDF（inverse document frequency）である [14]．本論文では，総文書数を 200 億ページと見なして IDF を算出する．一般的な語であるほど DF が増えるため，IDF は減少する．具体的には，単語 w についての WebIDF 値を $I(w)$ とするとき，閾値 I_1，I_2 を用いて

$$s_{\mathrm{webidf}}(w) = \begin{cases} -1.0 & (I(w) < I_1) \\ -(1 - x^2)^2 & (I_1 \le I(w) \le I_2) \\ 0 & (I(w) > I_2) \end{cases}$$

$$x = \frac{I(w) - I_1}{I_2 - I_1}$$

である．本論文では，$I_1 = 2.5$，$I_2 = 5$ を用いた．

- s_{sem}：意味クラス（(h)，(i) に対応）
役職名や分類名のように，固有表現ではあるが一般名詞である意味クラスの場合に負のスコアを与える．本論文では，役職名の場合は −0.5，分類名の場合は −0.2 を与える．

各候補ワードについて，上述の属性スコアの重み付き線形和としてワードスコアが算出される．それぞれの属性スコアに対する重みは，実際のデータに基づき決定する．本論文では，5 章で述べるとおり，調査結果に基づき実験計画法および焼きなまし法 [1] を用いて決定した．

最終的に，高いワードスコアが付与された単語が検索キーワードとして出力される．3 章で紹介したアプリケーションでは，検索キーワード抽出ステップが終了すると，通知アイコンがハイライトされる．ユーザがアイコンをタッチすると検索キーワード一覧が表示される．このアプリケーションでは，1 画面あたり 8 個の検索キーワードが表示される．

4.4 関連情報検索

ユーザが検索キーワードの 1 つをタッチすると，Web 検索が実行される．検索対象のジャンルやサービスは各検索キーワードごとのボタンの中に表示される．それぞれのジャンルやサービスの切替えは検索キーワードや特定ドメイン指定の追加により実現される．たとえば図 1 のアプリケーションの場合，単純なキーワード検索，ニュース検索，Wikipedia 検索が利用可能である．

また，各検索キーワードには意味クラスが付与されてい

るため，意味クラスに応じた検索を実行することも可能である．たとえば，人名であればプロフィールや画像を優先して検索するといった用途が考えられる．あるいは，クエリフリー型の検索支援 UI [27] を用いる場合，検索対象のコンテンツに適した意味クラスの単語を用いることも可能である．

5. 抽出対象となる単語の調査

アプリケーションの実装に際して，4.3 節で述べた属性スコアの重みを決定するために検索クエリとして用いられやすい単語の種類や属性を調査した．

モバイル端末で 2 日に 1 回以上 Web ページを閲覧する 20 代から 50 代の 50 名（男性 35 名，女性 15 名．アルバイトとしてモニタ調査に応募）を対象に，検索キーワードおよび検索サービスをそれぞれリストから選択するスマートフォン上の試作 UI を用いて各自約 40 分自由利用することで，検索に用いられやすい単語を調査した．利用環境は屋内で，アプリケーションの機能を説明したうえで，自由に Web 閲覧し，調べたいことがある場合に提案機能を試すよう教示した．

検索サービスとしては，通常のキーワード検索に加え，「Wikipedia」「動画」「ニュース」「ブログ」「クチコミ（商品の評判）」の 5 種類の検索サービスを用意した．調査の後，各被験者に対し 5 段階評価でサービスをどの程度利用したいかアンケート入力してもらった（5：いつも使う，4：よく使う，3：時々使う，2：あまり使わない，1：使わない）．

調査により行われた合計 407 回の検索において用いられた上位 5 種類の意味クラスとその頻度を**表 1** に示す．この結果を受け，4.3 節のスコアリングにおいて，これらの単語の重み付けを高くした．

図 4 は利用したい検索サービスについてのアンケート結果である．図 4 より，ニュース，Wikipedia，動画が被験者に好まれることが分かる．以降の実験では，通常のキーワード検索に加え，要望の大きいニュース，Wikipedia の 2 種類を用いた．

次に，候補ワードのスコアリングにおいて単語のどのような特徴が重要であるかを調査するため，エンターテインメント，文化，スポーツ，科学技術，政治，経済，グルメ，旅行，国際ニュース，生活の各ジャンルについて 151 URL の日本語 Web ページを対象に「正解」ワードを収集した．

収集されたページについて，主に著者らの組織に所属する IT 系技術者 29 名を対象とし，Web ページのスナップショットを配布した．「正解」の抽出方法としては，Web ページのスナップショットを PC 画面に表示してマーキング，あるいは紙に印刷してペンでマーキングするよう教示した．その結果，延べ 2,310 単語が抽出された．平均すると，1 URL あたり 4.98 名が作業を行い，10.56 種類の単語が選択されている．この結果を参考に，検索キーワード抽出処理の重み付けを調整するとともに，主に 3 名以上が選択した単語を後述のクローズド評価用の正解として登録した．

表 1 検索クエリとして利用された意味分類（上位 5 件）
Table 1 Semantic class used for search queries (top 5).

分類	頻度
人物	106
無生物（商品）	83
組織	81
無形物（タイトル，事件）	60
地域	42

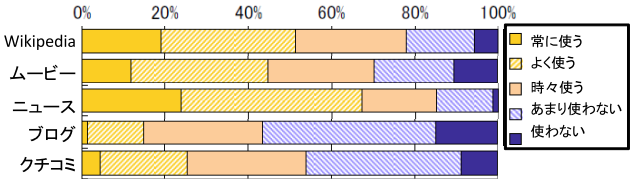

図 4 必要な検索サービス
Fig. 4 Necessary search services.

その後，4.3 節であげたどの特徴が検索キーワードとして重要であるかを調査した．まず，主効果と交互作用を調査するため，実験計画法による 2^k 要因実験を実施した．1/2 部分要因実験（分解能 VI）で 32 試行の実験を行った結果，単語の出現位置と WebIDF の交互作用，単語の出自と単語の出現位置の交互作用による効果が大きく，出現頻度や補足説明の有無による効果は相対的に小さいことが分かった．そこで，出現頻度や補足説明の有無に関する係数を固定したうえで，残りのパラメータを焼きなまし法 [1] により最適化する方式をとった．

最終的に決定された具体的な重みについては，後述の 7.1 節において学習データおよびテストデータすべてを含めて算出した数値を報告する．

6. 実装

第 1 検索キーワードを抽出するためのパラメータを決定した後，提案手法を備えたモバイル検索アプリケーションを Web ブラウザのアドオンとして実装した．Web ブラウザが HTML を読み込むとアドオンが呼び出され，HTML の DOM（documnet object model）ツリー文字列が検索キーワード抽出モジュールに渡される．前章で述べた検索キーワード抽出処理処理が完了すると，アイコンにより通知される．

検索キーワード抽出モジュールは Windows® XP 以降，Windows Mobile® 6.1 以降，Linux，Android 2.1 以降の各プラットフォームにおいて実装されている．本論文では，スマートフォン T-01A（OS: Windows Mobile® 6.5（シングルタッチ），CPU: 1 GHz，SDRAM 256 MB，4.1 インチタッチディスプレイ，ハードウェアキーボード非搭載）にお

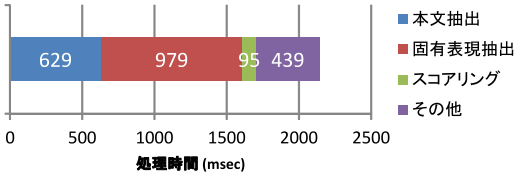

図 5 キーワード抽出の平均処理時間
Fig. 5 Average term-extraction processing time.

いて，Web ブラウザ Internet Explorer® Mobile のアドオンとして実装させたものを用いている．なお，本モジュールは，モジュール本体，辞書，ルールあわせて約 10 MB であり，他のデバイス，たとえば TV やカーナビゲーションシステムなどの組み込み機器でも利用可能である．

6.1 性能

検索キーワード抽出処理において必要な性能は「ユーザが検索しようとするタイミングでキーワードが抽出済みであること」であるといえる．つまり，ユーザが本文を読み終えるなど，まとまった分量の文を読むまでに抽出される必要がある．

ニュース，ブログなど複数分野の 15 URL（ページサイズは平均 89 k バイト）に対し，検索キーワード抽出処理でかかった時間を 5 回ずつ測定した結果の平均を**図 5** に示す．処理時間は，検索キーワード抽出全体で平均 2.14 秒であった．キーワード抽出処理はユーザが Web ページを閲覧中にバックグラウンドで実行されるため，ユーザにとっては関連情報を検索したいと思った際には，すでにキーワード抽出が終了していると考えられる．また実用する際には，長文ページなどで処理時間がかかってしまうのを防ぐため，閾値（10,000 文字）を超える長さのテキストでは本文抽出や固有表現抽出の途中で処理を打ち切っている．打ち切り処理を行う場合も，閾値は最終的にユーザに提示する検索キーワード数と必要な本文テキスト長より十分大きいため，ユーザから見た提示ワード数にはほとんど影響しない．

7. 検索キーワード抽出と情報検索の評価

提案方式が実用的に利用できるか確認するために，実装した検索支援アプリケーションの評価を行った．ここで，「実用的である」とは，「自分で選択または入力してキーワード検索するよりも早く検索が行える程度に適切なキーワードを提示できること」を指す．

以下，クローズド評価による検索キーワード抽出精度，およびオープン評価による検索キーワード抽出精度および情報検索の評価について述べる．

7.1 クローズド評価

本節では，提案した検索キーワード抽出手法のクローズド評価について述べる．

評価対象として，前章で述べた様々なジャンルから集められた 148 URL の Web ページを用いた．登録された正解ワードは 410 個（1 URL あたり平均 2.77 個）で，作成に際し 2 名以上が登録・確認を行っている．また，本文抽出の精度評価のために，ページの主要コンテンツと呼べる文章を本文の正解として登録した．

評価指標としては，以下の指標を用いた．

- 本文抽出 F 値：正解として登録した本文に対する，本文抽出結果の文字数比率による F 値
- 検索キーワード抽出再現率：正解として登録した検索キーワードの再現率
- 検索キーワード抽出 8 位再現率：正解として登録した検索キーワードの，抽出された検索キーワードの上位 8 件以内における再現率

本評価では，多くの場合出力可能な検索キーワード個数より正解個数の方が少ないため，正解が含まれるかどうかを重視するために，適合率ではなく再現率を用いた．

評価結果は以下のとおりである．数値はいずれも平均値である．

- 本文抽出 F 値：0.874
- 検索キーワード抽出再現率：0.749
- 検索キーワード抽出 8 位再現率：0.622

1 URL あたり平均して 2.77 個正解が登録されていることから，図 1 (b) の UI を用いる場合，最初の画面に表示される 8 個のクエリワードのうち平均して 1，2 個の正解ワードが提示されることになる．検索キーワード抽出再現率と 8 位再現率の差が 0.127 と比率が小さく，平均してワード数の差は 1 個未満であることから，スコアリングによるクエリワードの取りこぼしはほとんど起こっていないことが分かる．また，候補ワード抽出処理において正解ワードが問題なく抽出されていることから，本文抽出処理はおおむね成功しているといえる．

しかしながら，候補ワード抽出において約 25%が抽出に失敗している．原因としては，スコアリング段階での失敗，固有表現抽出方式で扱える語彙の不十分さ，単語の表記自体は一般語と変わらない場合，などの原因があり，これらの解消は今後の課題である．

最終的に，合計 299 URL の正解データを用いて焼き鈍しを実施した結果，4.3 節で述べた各属性スコアの重みは

$$8.9s_{\mathrm{origin}} + 7.5s_{\mathrm{sem}} + 2.5s_{\mathrm{header}} + 2.05s_{\mathrm{position}} + 1.6s_{\mathrm{webidf}} + 0.25s_{\mathrm{semfreq}}$$

となった．それぞれの重みにはばらつきがあるが，極端に小さな係数は含まれないため，それぞれの属性スコアが最終スコアに影響を与えることが分かる．

7.2 オープン評価

クローズド評価に続き，実際のユーザによる印象を調査するために提案インタフェースの検索キーワード抽出に関するオープン評価を実施した．

被験者は著者らの組織に所属する IT 系技術者 14 名である．各被験者は屋内のオフィス環境でモバイル端末（T-01A）を用い，約 1 時間ずつ，ふだん PC や携帯端末で閲覧するのと同様，自由に Web 閲覧を行った．アプリケーションの機能を説明したうえで，自由に Web 閲覧し，調べたいことがある場合に提案機能を試すよう教示した．Web ブラウザ起動時に最初に表示するページは，内容に関し知らない事柄が含まれると考えられるニュースサイトとした．本実験においては「Yahoo!ニュース[*3]」を用いた．また，本評価では，検索エンジンとして Google[*4] を用いた．

実験手順は以下のとおりである．

(1) 被験者は約 1 時間，自由に Web ブラウジングを行う．関連情報を検索したいタイミングで通知アイコンを押し，検索支援機能を利用する．検索に利用したい単語が表示されている場合は選択して検索し，表示されていない場合は画面上の検索ボックスから検索キーワードを自分で入力して検索する．

(2) 検索結果の提示は図 1 (c) の UI ではなく，評価用 UI（**図 6**）を用いて行われる．提示件数は最大 8 件とし，被験者は検索するごとに以下の評価を入力する．

検索目的 被験者が知りたいことが明確であったかどうかを確認するため，「検索目的」について「あった」（能動的検索），「なんとなく」（受動的検索），「間違えた」（間違えてボタンを押した場合）の 3 種類から選択する．「間違えた」は操作ミスのため，集計からは除外した．

各検索結果に対する評価値 それぞれの検索結果に対し，被験者の満足度合いに応じて 3 段階で評価する．評価値は以下のとおりである．

3: 見たい（そのページの内容を閲覧したい）
2: 出てもよい（閲覧したいわけではないが，検索結果に含まれていてもよい）
1: 見たくない（結果に表示されてほしくない）

(3) 自由利用の後，検索支援機能を用いたときに提示された検索キーワード一覧を確認し，それぞれの単語に対する満足度を以下の 3 段階で評価した．

3: 検索したい
2: 検索キーワード一覧に出てもよい
1: 検索キーワード一覧に不要

操作フローのまとめを，**図 7** に示す．ユーザは通常の Web ブラウジング中の任意のタイミングで検索支援機能を呼び出す．通知アイコンを押す操作 (A) はユーザが関連情報を調べたいと思った回数と見なすことができる．通知アイコンを押すとキーワード一覧が表示され，ユーザはキーワードを選択することで検索を行う (B)．このときに所望のキーワードが提示されなかった場合には，自分でキーワードを入力して検索することも可能である (B')．キーワード選択 (B) もしくはキーワード入力 (B') を行うと検索結果一覧が表示される．ユーザは結果一覧から選択する ((B)，(B') に対する選択がそれぞれ (C)，(C')）ことで次のページを閲覧することができる．

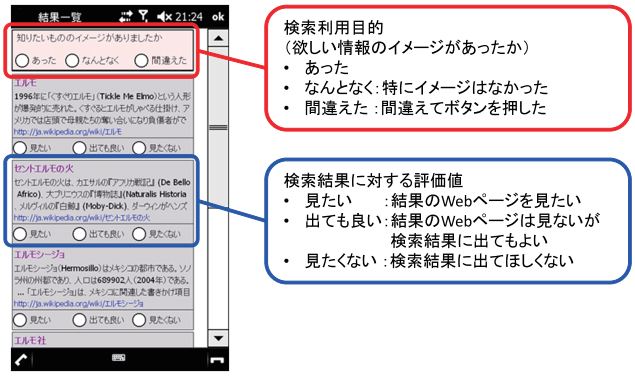

図 6 オープン評価用 UI
Fig. 6 UI for open evaluation.

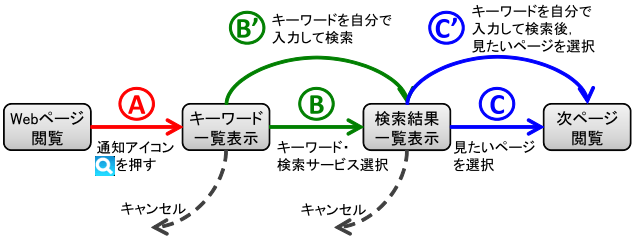

図 7 操作フロー
Fig. 7 Operation flow.

オープン評価では，以下の方針で評価指標を定義した．

提案機能はユーザが閲覧中のページの関連情報を調べることを目的とすることから，ユーザが関連情報を調べたいときに，どの程度満足できる推薦結果を提示できるかを測る指標として，以下の式で表されるユーザセッション受容率を定義する．

$$\text{ユーザセッション受容率} = \frac{\text{検索結果の受容セッション数 (C)}}{\text{検索支援機能利用セッション数 (A)}}$$

ここで，1 セッションは通知アイコンを押してから次のページに遷移するまでを表す．セッション受容とは，少なくとも 1 つの検索結果が「3：検索したい」と評価されることを意味する．

また，検索したい検索キーワードが提示されている割合を測定するため提示キーワード利用率を，検索結果が受容される割合を測定するため検索結果セッション受容率をそれぞれ以下の式により定義する．

[*3] http://headlines.yahoo.co.jp/hl
[*4] http://www.google.co.jp/

表 2 オープン評価の結果（各数値はユーザごとの平均）
Table 2 Result of open evaluation (each value is the average of users).

項目	平均
閲覧ページ数	40
検索支援機能利用回数 (A)	13.9
検索キーワード選択セッション数 (B)	10.2
B のうち能動的検索の場合 (B*)	6.57
検索クエリをユーザ自身が入力した回数 (B')	1.7
検索セッション数 (B+B')	11.9
検索キーワード提示後のキャンセル数	2.0
検索キーワード選択後の受容セッション数 (C)	8.36
受容されたセッション数 (C+C')	9.99
検索結果表示後のキャンセル数	1.91
ユーザセッション受容率 (C/A)	62.8%
提示ワード利用率 (B/A)	73.2%
検索結果セッション受容率 (C/B)	84.7%

$$\text{提示キーワード利用率} = \frac{\text{キーワード選択による検索セッション数 (B)}}{\text{検索支援利用セッション数 (A)}}$$

$$\text{検索結果セッション受容率} = \frac{\text{検索結果の受容セッション数 (C)}}{\text{キーワード選択による検索セッション数 (B)}}$$

オープン評価の結果を**表 2** に示す．表 2 より，平均して 73.3%の検索において提示キーワードが選択されていることが分かる．また，検索キーワード選択後は検索セッションの 84.7%について受容されている．つまり，62.8%の検索セッションにおいては文字入力することなくタッチ操作だけで次の Web ページにユーザは進んでおり，その分だけ手間を削減できたといえる．

このうち，検索目的による効果の違いについて述べる．上述のとおり，アプリケーション利用者が検索する場合，能動的検索と受動的検索の 2 種類の検索目的がある．このうち，通常のキーワード検索と同じ検索目的といえるのは能動的検索である．**表 3** に検索の利用目的別の受容率を示す．表 3 より，受動的検索の場合（70.7%）と比べ能動的検索の場合受容率は 93.0%と高いことが分かる．

これは，受動的検索の場合調べたい情報が予想されているというよりは「なんとなく面白い Web ページ」を期待することから，検索結果において「面白い」ページを見つけられなかったときに評価値が下がると考えられる．また，候補ワードを表示した後のキャンセル操作は，受動的検索の場合「どんな単語が表示されるか試しに見る」操作の失敗と考えることができる．ユーザが明確な意図を持っている場合は，適切な検索ワードを見つけることができなかった場合自分で検索クエリを入力する．

ユーザが能動的検索を行う場合の提示ワード利用率は $B^*/(B^*+B') = 79.4\%$，つまり約 5 回に 1 回程度は提示ワードが利用されないことが分かる．抽出された検索キーワードを選択するだけのアプリケーションの場合，この精度ではユーザは満足しない可能性があるが，本アプリケーションの場合，提示ワードを見つけられない場合は即座にテキスト入力による検索を利用できるため，アプリケーション全体として見た場合，検索するための手間を削減する手段としては効果があると考えられる．

表 3 検索利用目的ごとの検索結果セッション受容率の比較
Table 3 Session acceptability for each search intention.

項目	能動的検索	受動的検索
検索キーワード選択セッション数	6.57	3.64
検索結果セッション受容率 (%)	93.0	70.7

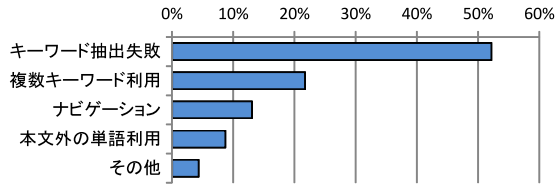

図 8 被験者自身がクエリを入力した理由
Fig. 8 Reason for manual query input.

表 4 検索キーワード選択後の検索サービスごとの検索結果セッション受容率
Table 4 Session acceptability of search results for each search service after selecting displayed candidate terms.

項目	検索セッション数	検索結果セッション受容率
キーワード検索	5.29	0.884
ニュース検索	2.64	0.761
Wikipedia 検索	2.29	0.803

なお，平均して 1 人あたり 1.7 回のクエリ入力が行われているが，これはユーザが明確に検索する意図があったにもかかわらず 1.7 回のセッションでは適切な検索キーワードを見つけられなかったことを意味する．評価後，被験者にクエリを自分で入力した理由を確認した結果の分布を**図 8** に示す．主な理由は「検索キーワード抽出失敗」である．これは今後抽出精度の改良により対応する必要がある．また，20%は 2 個またはそれ以上の検索キーワードが用いられている．これに対応するためには，たとえば 2 章で紹介した他のクエリ拡張方式を組み合わせる必要がある．10%は，たとえばニュースヘッドラインのページを探すために 'Yahoo' と検索するような新しい話題のページへのナビゲーション利用である．

また，**表 4** に検索ジャンルごとの検索結果セッション受容率の比較を示す．表 4 より，ニュース検索および Wikipedia 検索の検索結果セッション受容率はキーワード検索の場合よりも低いことが分かる．これは，これら 2 種類の検索によって得られる検索結果の数がキーワード検索の場合よりも少なく，ユーザが満足する結果を得られる場合が減ることが原因として考えられる．

ここまでの評価を通じて，提案した 1 番目の検索キーワード抽出手法は情報検索場面の多くで有用であると結論付けることができる．

また，オープン評価で検索されたユニークな単語 160 語について，固有表現抽出処理のうちルールによる出力か辞書による出力かを比較したところ，少なくとも 107 個（66.9%）はルールを用いた出力であった．本論文では継続的な評価について述べていないものの，アプリケーションで辞書のアップデートができず，すべて新語に置きかわったとしても 3 回中 2 回程度は検索キーワード候補が抽出できるといえる．

8. おわりに

本論文では，閲覧 Web ページに関連する情報の検索を支援するための第 1 検索キーワード抽出手法を提案した．提案手法は，HTML 文書からの本文抽出，固有表現抽出を用いた候補ワード抽出，単語の特徴を考慮したスコアリングにより構成される．

50 名の被験者による調査を通じ，どのような単語が検索キーワードとして適切か調査するとともに，29 名による 151 URL への入力に基づき検索されやすい単語の特徴について調査した．調査結果をふまえ，2，3 回のタッチ操作だけで関連情報を検索可能とするスマートフォン向けワンタッチ検索インタフェースを実装し，299 URL の Web ページを用いたクローズド評価と 14 名の被験者によるオープン評価を通じて実用的な性能・精度で利用できることを確認した．特に，ユーザが明確な検索意図を持っている場合に提案手法がユーザの負担を軽減する効果を確認した．

今後の課題の 1 つは，提案手法の改良である．テキストボックスを利用するなどの代替手段が用意されないアプリケーションで利用するには検索キーワード抽出精度は十分とはいえず，今後も引き続き改良する必要がある．

現在の手法では，HTML 文書全体を処理対象としているが，実際には画面に描画されていない領域に対しても検索キーワード抽出処理が行われている．実際に閲覧された領域を考慮し，閲覧されていない単語は抽出しないなどの工夫をすることで適合率を向上させることができると考えられる．もちろん，本論文では対象としなかった 2 番目以降の検索キーワードを提示する手法を組み合わせることで様々な用途の検索クエリに対応できるようになる．ユーザの Web 閲覧履歴や検索履歴を活用したスコアリングの改良・学習方式も考えられる．

他の課題としては，検索利用手順の改良・拡充があげられる．たとえば，提案アプリケーションでは選択可能なカテゴリはあらかじめ決められているが，単語によってはカテゴリを選ぶことに意味がない場合がある．単語ごとに「調べる意味のあるカテゴリ」を切り替えて表示する機能が考えられる．利用するまでの流れに関して，本論文では，文章を読んだ後に検索する，という流れを念頭においたアプリケーションについて述べたが，読んでいる途中で検索したい単語を直接タッチする方式や，ページ内に元々用意されている「関連ページへのリンク」を積極的に表示する方式も考えられる．

また，本論文ではモバイル端末の入力デバイスや画面サイズの制約を主な課題としてあげたが，ユーザのおかれている状況を利用 [28] することも考えられる．モバイル用途では，外出時の位置情報の利用 [3] や外出時の情報ニーズ [5] を考慮した方式の併用も考えられる．

参考文献

[1] Aarts, E. and Korst, J.: *Simulated Annealing and Boltzmann Machines*, John Wiley and Sons Inc., New York, NY (1988).

[2] Antonellis, I., Garcia-Molina, H. and Chang, C.: Simrank++: Query Rewriting through Link Analysis of the Clickgraph, *Proc. WWW2008*, pp.1177–1178, ACM (2008).

[3] Bellotti, V., Begole, B., Chi, E., Ducheneaut, N., Fang, J., Isaacs, E., King, T., Newman, M., Partridge, K., Price, B., Rasmussen, P., Roberts, M., Schiano, D. and Walendowski, A.: Activity-based Serendipitous Recommendations with the Magitti Mobile Leisure Guide, *Proc. CHI2008*, pp.1157–1166, ACM (2008).

[4] Borthwick, A.: A Maximum Entropy Approach to Named Entity Recognition, Ph.D. Thesis, New York University, New York, NY (1999).

[5] Church, K. and Smyth, B.: Understanding the Intent Behind Mobile Information Needs, *Proc. IUI2009*, pp.247–256, ACM (2009).

[6] Cui, H., Wen, J., Nie, J. and Ma, W.: Query Expansion by Mining User Logs, *IEEE Trans. Knowl. and Data Eng.*, Vol.15, No.4, pp.829–839 (2003).

[7] DeJong, G.: An Overview of the FRUMP System, *Strategies for Natural Language Processing*, Lehnert, W. and Ringle, M. (Eds.), chapter 5, pp.149–176, Lawrence Erlbaum and Associates, Hillsdale, NJ (1982).

[8] Hart, P. and Graham, J.: Query-Free Information Retrieval, *IEEE Expert*, Vol.12, No.5, pp.32–37 (1997).

[9] He, Q., Jiang, D., Liao, Z., Hoi, S., Chang, K., Lim, E. and Li, H.: Web Query Recommendation via Sequential Query Prediction, *Proc. ICDE2009*, pp.1443–1454, IEEE (2009).

[10] Henzinger, M., Chang, B., Milch, B. and Brin, S.: Query-Free News Search, *World Wide Web*, Vol.8, No.2, pp.101–126 (2005).

[11] Hobbs, J., Appelt, J., Bear, J., Kameyama, M. and Tyson, M.: FASTUS: A System for Extracting Information from Text, *Proc. Workshop on Human Language Technology*, pp.133–177 (1993).

[12] Kamvar, M., Kellar, M., Patel, R. and Xu, Y.: Computers and iPhones and Mobile Phones, oh my!: A logs-based comparison of search users on different devices, *Proc. WWW2009*, pp.801–810, ACM (2009).

[13] Komaki, D., Ohnishi, K., Arase, Y., Hattori, G., Hara, T. and Nishio, S.: Design and Implementation of A Click-Search Interface for Web Browsing Using Cellular Phones, *Intl. J. Web and Grid Services*, Vol.5, No.1, pp.66–84 (2009).

[14] Kondo, M., Tanaka, A. and Uchiyama, T.: Search Your

Interests Everywhere!: Wikipedia-based Keyphrase Extraction from Web Browsing History, *Proc. HT2010*, pp.295–296, ACM (2010).
[15] Kraft, R., Chang, C., Maghoul, F. and Kumar, R.: Searching with Context, *Proc. WWW2006*, pp.477–486, ACM (2006).
[16] Kraft, R., Maghoul, F. and Chang, C.: Y!Q: Contextual Search at the Point of Inspiration, *Proc. CIKM2005*, pp.816–823, ACM (2005).
[17] Lieberman, H.: Letizia: An Agent That Assists Web Browsing, *Proc. IJCAI1995*, Vol.1, pp.924–929, Morgan Kaufmann (1995).
[18] Liu, B., Chiticariu, L., Jagadish, H. and Reiss, F.: Automatic Rule Refinement for Information Extraction, *Proc. VLDB Endow.*, Vol.3, No.1-2, pp.588–597 (2010).
[19] Ma, Q. and Tanaka, K.: Topic-Structure-Based Complementary Information Retrieval and Its Application, *ACM Trans. Asian Language Information Processing*, Vol.4, No.4, pp.475–503 (1995).
[20] Menemenis, F., Papadopoulos, S., Bratu, B., Waddington, S. and Kompatsiaris, Y.: AQUAM: Automatic Query Formulation Architecture for Mobile Applications, *Proc. MUM2008*, pp.32–39, ACM (2008).
[21] Okamoto, M., Kikuchi, M. and Yamasaki, T.: One-button Search Extracts Wider Interests: An Empirical Study with Video Bookmarking Search, *Proc. SIGIR2008*, pp.779–780, ACM (2008).
[22] Okamoto, M., Watanabe, N., Kikuchi, M., Iida, T., Sasaki, K., Horiuchi, K., Yamasaki, T., Omura, S. and Hattori, M.: First Query Term Extraction from Current Webpage for Mobile Applications, *Proc. MUM2010*, pp.19:1–19:9, ACM (2010).
[23] Rahurkar, M. and Cucerzan, S.: Predicting When Browsing Context is Relevant to Search, *Proc. SIGIR2008*, pp.841–842, ACM (2008).
[24] Sakai, T.: Advanced Technologies for Information Access, *Intl. J. Computer Processing of Oriental Languages*, Vol.18, No.2, pp.95–113 (2005).
[25] Sakai, T., Saito, Y., Ichimura, Y., Koyama, M., Kokubu, T. and Manabe, T.: ASKMi: A Japanese Question Answering System based on Semantic Role Analysis, *Proc. RIAO2004*, pp.215–231 (2004).
[26] Sarma, A., Jain, A. and Srivastava, D.: I4E: Interactive Investigation of Iterative Information Extraction, *Proc. SIGMOD2010*, pp.795–806, ACM (2010).
[27] Watanabe, N., Okamoto, M., Kikuchi, M., Iida, T., Sasaki, K., Horiuchi, K. and Hattori, M.: Designing Mobile Search Interface with Query Term Extraction, *Proc. KES2011, Part III*, pp.67–76, Springer-Verlag (2011).
[28] White, R., Bailey, P. and Chen, L.: Predicting User Interests from Contextual Information, *Proc. SIGIR2009*, pp.363–370, ACM (2009).
[29] Yih, W., Goodman, J. and Carvalho, V.: Finding Advertising Keywords on Web Pages, *Proc. WWW2006*, pp.213–222, ACM (2006).
[30] Yoshida, T., Nakamura, S. and Tanaka, K.: WebBrowSearch: Toward Web Browser with Autonomous Search, *Proc. WISE2007*, pp.135–146, Springer-Verlag (2007).

渡辺 奈夕子 （正会員）

2004 年東京大学理学部情報科学科卒業．2006 年同大学大学院情報理工学系研究科修士課程修了．同年（株）東芝入社．現在，同社研究開発センター知識メディアラボラトリー所属．主に情報推薦・検索システム，ユーザインタフェースの研究開発に従事．

岡本 昌之 （正会員）

1998 年京都大学工学部情報工学科卒業．2003 年同大学大学院情報学研究科社会情報学専攻博士後期課程修了．同年（株）東芝入社．現在，同社研究開発センター知識メディアラボラトリー研究主務．博士（情報学）．主にコンテクストアウェア技術および情報抽出技術の研究開発に従事．2008 年日本データベース学会/電子情報通信学会データ工学研究会/情報処理学会データベースシステム研究会優秀若手研究者賞受賞．人工知能学会，ACM 各会員．

菊池 匡晃

2006 年大阪大学大学院工学研究科知能・機能創成工学専攻修士課程修了．同年（株）東芝入社．現在，同社デジタルプロダクツ&サービス社プラットフォーム&ソリューション開発センター所属．主にコンテクストアウェア技術，情報抽出技術，機器連携応用技術の研究開発に従事．2009 年日本ロボット学会論文賞受賞．

飯田 貴之

2005 年新潟大学工学部電気電子工学科卒業．2007 年同大学大学院自然科学研究科数理情報電子工学専攻博士前期課程修了．同年（株）東芝入社．現在，同社デジタルプロダクツ&サービス社プラットフォーム&ソリューション開発センター所属．主にモバイル分野におけるユーザ支援アプリケーションの開発に従事．

佐々木 健太

2007 年慶應義塾大学理工学部生命情報学科卒業．2009 年同大学大学院理工学研究科基礎理工学専攻博士前期課程修了．同年（株）東芝入社．現在，同社研究開発センター知識メディアラボラトリー所属．主にマイクロブログを対象としたユーザ行動分析の研究開発に従事．

堀内 健介

2004 年京都大学工学部電気電子工学科卒業．2006 年同大学大学院工学研究科電気工学専攻修士課程修了．同年（株）東芝入社．現在，同社デジタルプロダクツ&サービス社プラットフォーム&ソリューション開発センタープラットフォーム・ソリューション設計第三部所属．主にモバイル分野におけるユーザ支援アプリケーションの開発に従事．電子情報通信学会会員．

山崎 智弘

2000 年東京大学理学部情報科学科卒業．2002 年同大学大学院理学系研究科情報科学専攻修士課程修了．同年（株）東芝入社．現在，同社研究開発センター知識メディアラボラトリー研究主務．自然言語処理の研究開発に従事．電子情報通信学会会員．

大村 寿美 （正会員）

1993 年慶應義塾大学理工学部計測工学科卒業．1995 年同大学大学院理工学研究科計測工学専攻修士課程修了．同年（株）東芝入社．現在，同社デジタルプロダクツ&サービス社プラットフォーム&ソリューション開発センタープラットフォーム・ソリューション開発第五部第三担当グループ長．主にモバイル分野におけるユーザ支援アプリケーションの開発に従事．

服部 正典 （正会員）

1996 年九州大学大学院工学研究科情報工学専攻修了．同年（株）東芝入社．現在，同社研究開発センター知識メディアラボラトリー主任研究員．博士（情報科学）．2005～2006 年マサチューセツ工科大学客員研究員．主にエージェント技術，コンテクストアウェア技術の研究開発に従事．