小型情報端末向けWeb情報収集支援技術に関する研究

提出先　　大阪大学大学院情報科学研究科

提出年月　2009年1月

服部　元

関連発表論文

1. 学会論文誌発表論文

1. Satoshi Nishiyama, Gen Hattori, Chihiro Ono, and Hiroki Horiuchi: Lightweight FIPA Compliant Agent Platform on Java-Enabled Mobile Phone for Ubiquitous Services, *Journal of IPSJ*, Vol. 45, No. 2, pp. 575–585 (Feb. 2004).

2. 服部 元, 松本一則, 菅谷史昭: タグの深さを利用したコンテンツ間距離に基づく Web ページの自動分割方式, 日本データベース学会 Letters, Vol. 4, No. 1, pp. 149–152 (June 2005).

3. 服部 元, 松本一則, 菅谷史昭: 表形式情報集約のための連想性の高いオブジェクトラベルの自動抽出方式, 電子情報通信学会論文誌, Vol. J88-DI, No. 9, pp. 1467–1476 (Sept. 2005).

4. 服部 元, 松本一則, 菅谷史昭: コンテンツ間距離の標準偏差に基づく Web ページ動的分割方式, 情報処理学会論文誌：データベース, Vol. 47, No. SIG 8 (TOD 30), pp. 81–89 (June 2006).

5. 大槻一博, 服部 元, 星野春男, 松本一則, 菅谷史昭: 携帯向けオンラインニュース配信のための視聴/非視聴履歴に基づく嗜好クラスタ管理手法, 日本データベース学会 Letters, Vol. 6, No. 1, pp. 37–40 (June 2007).

6. 大槻一博, 服部 元, 松本一則, 滝嶋康弘, 菅谷史昭, 鹿喰善明: パーソナル・オンラインニュース配信システムの実証実験, 日本データベース学会論文誌, Vol. 7, No. 1, pp. 43–48 (June 2008).

7. 服部 元, 原 隆浩, 滝嶋康弘, 西尾章治郎: 周辺語を活用したクリック型 Web 検索システムの提案と評価, 情報処理学会論文誌：データベース, Vol. 1, No. 2, pp. 26–37 (Sept. 2008).

8. Daijiro Komaki, Kenji Ohnishi, Yuki Arase, Gen Hattori, Takahiro Hara, and Shojiro Nishio: Design and Implementation of a Click-Search Interface for Web Browsing Using Cellular Phones, *International Journal Web and Grid Services* (2008, to appear).

2. 国際会議

1. Gen Hattori, Chihiro Ono, Satoshi Nishiyama, and Hiroki Horiuchi: Making Java-enabled Mobile Phone As Ubiquitous Terminal by Lightweight FIPA Compliant Agent Platform, in *Proceedings of IEEE International Conference on Pervasive Computing and Communincations* (*PerCom 2003*), pp. 553–556 (Mar. 2003).

2. Gen Hattori, Keiichiro Hoashi, Kazunori Matsumoto, and Fumiaki Sugaya: Robust Web Page Segmentation for Mobile Terminal Using Content-Distances and Page Layout Information, in *Proceedings of International World Wide Web Conference* (*WWW 2007*), pp. 361–370 (May 2007).

3. Daijiro Komaki, Kenji Ohnishi, Yuki Arase, Gen Hattori, Takahiro Hara, and Shojiro Nishio: A Click-Search Interface for Web Browsing Using Cellular Phones, in *Proceedings of International Wireless Internet Conference* (*WICON 2008*) (CD-ROM), No. 4857 (Nov. 2008).

3. 研究会等発表論文（査読付）

1. 服部 元, 松本一則, 菅谷史昭: モバイルエージェント間通信のためのハイブリッド型メッセージ配送方式, 情報処理学会シンポジウムシリーズ マルチメディア, 分散, 協調とモバイルシンポジウム (DICOMO 2004) 論文集, Vol. 2004, No. 1, pp. 169–174 (July 2004).

2. 服部 元, 松本一則, 菅谷史昭: 表形式情報集約のための人に優しい検索キーの自動抽出方式の検討, 合同エージェントワークショップ&シンポジウム (JAWS 2004) 論文集, Vol. 2004, No. 32, pp. 425–432 (Oct. 2004).

3. 服部 元, 松本一則, 菅谷史昭: タグの深さに基づくコンテンツ間距離を利用したWebページの自動分割方式, 電子情報通信学会データ工学ワークショップ (DEWS 2005) 論文集, *http://www.ieice.org/˜de/DEWS/DEWS2005/procs/papers/6A-o3.pdf* (Mar. 2005).

4. 服部 元, 松本一則, 菅谷史昭: コンテンツ間距離の分布に基づく Web ページ動的分割方式, 情報処理学会データベースと Web 情報システムに関するシンポジウム (DBWeb 2005) 論文集, Vol. 2005, No. 16, pp. 73–80 (Nov. 2005).

5. 大槻一博, 服部 元, 星野春男, 松本一則, 菅谷史昭: 携帯向けパーソナルオンラインニュース配信のための視聴/非視聴履歴に基づく嗜好クラスタ管理手法, 電子情報通信学会データ工学ワークショップ (DEWS 2007) 論文集, *http://www.ieice.org/˜de/DEWS/DEWS2007/pdf/c2-8.pdf* (Feb./Mar. 2007).

6. 服部 元, 原 隆浩, 菅谷史昭, 西尾章治郎: クリック型 Web 検索のための重要語推定方式, 情報処理学会データベースと Web 情報システムに関するシンポジウム (DBWeb 2007) 論文集 (CD-ROM), Vol. 2008, No. 1A-3 (Nov. 2007).

7. 大槻一博, 服部 元, 松本一則, 滝嶋康弘, 星野春男, 菅谷史昭: パーソナルオンラインニュース配信システムの実証実験, 電子情報通信学会データ工学ワークショップ (DEWS 2008) 論文集, *http://www.ieice.org/iss/de/DEWS/DEWS2008/proceedings/files/b2/b2-2.pdf* (Mar. 2008).

8. 小牧大治郎, 大西健史, 荒瀬由紀, 原 隆浩, 服部 元, 西尾章治郎: 携帯電話を用いた Web 閲覧のためのクリック検索インタフェース, マルチメディア, 分散, 協調とモバイルシンポジウム (DICOMO 2008) 論文集, Vol. 2008, No. 1, pp. 626–633 (July 2008).

4. その他の研究会等発表論文

1. 西山 智, 服部 元, 小野智弘, 堀内浩規: 携帯端末のための FIPA 準拠小型エージェントプラットフォームの実装, 情報科学技術フォーラム (FIT 2002) 講演論文集 (CD-ROM), No. M-16 (Sept. 2002).

2. 服部 元, 小野智弘, 西山 智, 堀内浩規: 携帯端末のための FIPA 準拠小型エージェントプラットフォームの評価, 情報科学技術フォーラム (FIT 2002) 講演論文集 (CD-ROM), No. M-17 (Sept. 2002).

3. 服部 元, 小野智弘, 西山 智, 堀内浩規: 携帯電話用小型エージェントプラットフォームのためのゲートウェイ機能の設計, 情報処理学会全国大会 講演論文集 (CD-ROM),

No. 1H-5 (Mar. 2003).

4. 服部 元, 小野智弘, 堀内浩規: モバイルエージェントプラットフォームのためのメッセージ配送方式の検討, 電子情報通信学会ソサイエティ大会 通信講演論文集, Vol. 2003, No. 2, p. 272 (Sept. 2003).

5. 服部 元, 小野智弘, 堀内浩規, 菅谷史昭: モバイルエージェント用メッセージ配送方式の評価, 電子情報通信学会総合大会 通信講演論文集, Vol. 2004, No. 2, p. 37 (Mar. 2004).

6. 渡辺伸吾, 西山 智, 服部 元, 小野智弘, 越塚 登, 坂村 健: 既存ルータ混在環境におけるモバイル IP ハンドオーバの高速・高信頼化の提案, 情報処理学会全国大会 講演論文集 (CD-ROM), No. 1H-2 (Mar. 2004).

7. 西山 智, 渡辺伸吾, 服部 元, 小野智弘, 越塚 登, 坂村 健: モバイル端末における応用の要求に応じた通信メディア使い分け方式の提案, 情報処理学会全国大会 講演論文集 (CD-ROM), No. 1H-3 (Mar. 2004).

8. 服部 元, 松本一則, 菅谷史昭, 小池 淳, 山田 満, 坂村 健: 統合 PDA 端末の開発 (3)〜携帯端末のための Web ページ自動分割, 電子情報通信学会総合大会 通信講演論文集, Vol. 2005, No. 1, p. 714 (Mar. 2005).

9. 大槻一博, 服部 元, 帆足啓一郎, 星野春男, 菅谷史昭: 携帯向けオンラインニュース配信のための視聴履歴に基づく嗜好クラスタ管理手法の検討, 電子情報通信学会研究報告 (Web インテリジェンスとインタラクション研究会 WI2-2006-6), Vol. 2006, No. 6, pp. 113–118 (July 2006).

10. 武吉朋也, 服部 元, 小野智弘, 滝嶋康弘: Web コンテンツ作成支援のためのリンク目的を意識したリンク先推薦手法, 情報科学技術フォーラム (FIT 2008) 講演論文集 (CD-ROM), No. D-009 (Sept. 2008).

以上

内容梗概

近年，Web 上の情報を閲覧する環境が広がっている．PC(Personal Computer) だけでなく携帯電話やゲーム機，カーナビゲーションシステム，STB(Set Top Box) などのさまざまな通信可能な端末 (以下，小型情報端末と呼ぶ) に Web ブラウザが搭載されるようになり，自宅やオフィスの机上だけでなく，テレビを見ながら，あるいは外出先や移動中でも Web 情報を活用できるようになってきた．一方，Web 上の情報が爆発的に増えており，必要な情報を探し出すことが困難になりつつある．しかしながら，ほとんどの Web ページは PC 向けに作られているため，小型情報端末では使い勝手が悪い．例えば Web ページを表示する際に低解像度の画面では概観できないため，目的の情報にたどり着くまでに時間がかかってしまう．また，キーボードがないため検索キーワードを入力するのに手間がかかってしまう．そのため，Web 情報収集の自動化と効率化，ならびに小型情報端末に合った Web 検索方法の提案など，Web における情報収集作業を支援するシステムが必要であると考えられる．

そこで本論文では，PC 向けの Web ページの情報を小型情報端末で閲覧・検索する作業を支援する情報収集支援手法について検討する．ユーザの利便性に関する具体的な 3 つの課題に着目し，それぞれの解決策の提案と，システムの実装ならびに評価実験を行う．ここで 3 つの課題とは，(1) 特定の情報を時系列観測する作業におけるユーザの負荷を減らすこと，(2) 低解像度画面の端末で効率的に情報閲覧できるようにすること，(3) キーボードが使えない端末で検索操作を容易に行えるようにすること，である．

本論文は 5 章から構成され，その内容は次の通りである．まず，第 1 章において序論を述べる．

次に，第 2 章で，小型情報端末での特定の情報データを時系列に観測する作業におけるユーザの負荷を減らすことを目的とした，Web からの時系列情報の自動抽出手法の提案を行う．本手法は，ユーザが着目した任意の情報に対し，その Web ページ内で識別可能な ID(Identification) を自動設定することを可能し，ランキングが変動するなどにより Web ページ内の表示位置やその値が変わった場合でも，自動追跡が可能となる．ユーザは，対象の情報を一度指定するだけで，時間変化する情報を追跡可能となる．

第 3 章では，小型情報端末の低解像度画面を使って PC 向けの Web ページから必要な情報を探し出す作業におけるユーザの負荷を減らすことを目的とした，Web ページ分割手法

の提案を行う．本手法は，ユーザが指定した任意の Web ページに対し，タグの深さに基づくタグ構造情報と，テーブル表現などから得られるレイアウト情報を利用して，ページ内の情報の意味的なつながりを高精度に推定し，複数の小さな Web ページに自動分割することができる．ユーザは，分割された小さな Web ページのリストから好みの情報を選択する簡単な操作で，目的の情報を閲覧することが可能となる．

第 4 章では，小型情報端末での Web 検索操作におけるユーザの負荷を減らすことを目的とした，ワンクリック検索手法の提案を行う．本手法は，表示中の Web ページ内にユーザが調べたい注目語があった場合において，注目語の周辺から，注目語を修飾する重要な語を抽出して検索キーワードを自動生成することで，内容が関連する検索結果を得ることができる．ユーザは，調べたい注目語をクリックするだけで，検索結果を得ることが可能となる．

第 5 章では，最後に本論文の成果を要約したのち，今後の検討課題について述べ，本論文のまとめとする．

目 次

第1章

序章

1.1 研究の背景

近年，Web 上に創出される情報量が爆発的に増加し続けている状況にある．IDC (International Data Corporation) は，世界中で生成されたデジタルデータが，2007 年の 1 年間で約 281 エクサバイト (2.25×10^{21} ビット, 1 バイトは 8 ビット) 増加したと報告している．さらに総デジタルデータ量は，2006 年から 2011 年のまでの 5 年間で 10 倍に増大すると予測している [37]．これらの大部分はデジタルカメラやカメラ付き携帯電話，防犯カメラ等のデバイスから得られる画像であることから，これらの中から Web 上に展開される情報も，同様に増加しているといえる．一方，Web を閲覧する端末については，PC(Personal Computer) よりも小型な携帯電話やゲーム機，STB(Set Top Box)，カーナビゲーションシステムなどのさまざまな通信可能な端末 (以下，小型情報端末と呼ぶ) に Web ブラウザが搭載されることが一般的になっている．特に携帯電話においては，2008 年 9 月の電気通信事業者協会の調査報告 [60] によると，国内の全契約端末の約 85.8%にあたる約 9000 万台が Web アクセス可能な端末となっている．このように，自宅やオフィスの机上だけでなく，テレビを見ながら，あるいは外出先や移動中でも Web 情報を収集し，活用できる環境が普及してきているといえる．

情報収集の対象となる Web ページの多くは，PC 向けに画面のレイアウトや情報量が調整されており，比較的大きな画面とキーボードやマウス等の自由度の高いユーザインタフェースがなければ容易に閲覧することはできない．さらに，検索エンジンを利用して，目的の情報を探し出す方法が一般的に行われているが，1 つの検索語を入力するだけでは数百万

件以上もの検索結果が返されることもあり，PC であってもすべてに目を通すことはできない．一方，小型情報端末向けの Web ページは，例えば携帯電話向けの Web ページが公開されているが，携帯電話は小さな画面や自由度の低い入力デバイス等，ユーザインタフェースの制限があることから，携帯電話向けの Web ページが持つ情報量は少なく調整される傾向にある．さらに携帯電話向けの Web ページの数は PC 向けと比較してはるかに少数である．そのため，小型情報端末を利用して情報量が豊富な PC 向けの Web ページを閲覧する要求が高まっている．実際，PC 向けの Web ページを表示可能なブラウザ [56] を搭載した携帯端末が市販されている．しかしながら，このようなブラウザは，元の Web ページのデザインやレイアウトをある程度保持することができる反面，小型情報端末の入力デバイスとして一般的な，方向キーや 10 キーがメインのテレビリモコンのような入力デバイス (以下，リモコン型 UI(User Interface) と呼ぶ) では，目的の情報が書かれた部分にたどり着くまでに多くのスクロール操作が必要となってしまい，ユーザの負担が大きい．さらに，PC の高解像度ディスプレイを利用した閲覧のように，Web ページ全体の情報を概観することができないため，目的の情報が表示されている部分に直線的に移動することは困難である．また，Web ページで提供される情報は，例えばランキングサイトや天気予報サイトなど，紙媒体の新聞や雑誌等よりも情報がリアルタイムに更新されるため，時間とともに目的の情報の値や表示位置が変化する場合もある．このとき，ユーザが最新情報を得るためには，頻繁に Web ページを訪問して表示位置を探す必要がある．このような継続的な情報収集作業は，特に小型情報端末の場合は Web 閲覧の困難さのために，ユーザの作業コストが膨大となってしまう．さらに，情報収集作業は，検索操作と閲覧操作の繰り返しであると考えられるが，リモコン型 UI では文字入力は PC のキーボードと比較すると自由に入力できないことから，多くの検索語を入力することは困難な作業となる．ここで，OneStat.com 社の調査結果によると [57]，Web 検索時に入力する検索語数で利用頻度が最も多いのは 2 語であり，また 10 語までの頻度分布で平均すると約 3 語と報告されている．よって，目的の情報を探すためには多くの文字を入力しなければならない．

以上により，小型情報端末を利用した Web における情報収集作業を効率化するシステムの実現が重要であり，検索操作と閲覧操作の両方について検討する必要がある．

1.2 アプローチ

本研究では，1.1 節で述べた Web 情報収集支援システムの実現を目的とする．まず研究方針を定めるため，既存研究を分類して本研究の位置づけを明確化し，研究項目を具体的に示す．

1.2.1 本研究の位置づけ

小型情報端末を利用した検索操作と閲覧操作に対する支援技術の研究方針として，本研究では，入力方式の改善と，表示方式の改善の 2 つの方針で取り組むこととする．これにより，Web 情報収集支援に関連する技術は大きく 4 つの研究分野に分類することができる．関連する既存研究と，本研究において検討する個々の研究項目の位置づけを図 1.1 に示す．

図 1.1 において，上半分が検索支援技術の領域となり，上に行くほど適用範囲の広い汎用的な技術，下に行くほど適用範囲が限定的となる具体性の高い技術である．一方，下半分が閲覧支援技術の領域であり，下に行くほど元の Web ページを忠実に再現することを重視して見やすくする技術，上に行くほど再現性を重視せずに表示する情報を絞り込むことで見やすくする技術である．また，右半分が操作方法の改善技術の領域であり，右に行くほどユーザの操作数が少なくなる技術である．一方，左半分が表示方法の改善技術の領域であり，左に行くほど小型端末上での見やすさを重視した技術である．ここで，小さい丸は既存研究の位置づけを表し，大きい楕円は本研究の個々の研究課題が目指すスコープを表す．破線の円は，本研究で扱わない領域を表す．

まず，左上の「表示方式の改善による検索支援技術」の領域においては，既存研究として，適用先の汎用性の高い順に，タグの木構造パターンを利用して情報抽出を行う方式 [13, 14, 21, 42, 61]，新旧の Web ページの差分を利用して情報抽出を行う方式 [52]，対話的分類学習を利用して逐次興じ学習により情報抽出を行う方式 [65] がある．しかしながら，これらの方式は，小型情報端末に適切な表示方法に関する検討は行っていない．そこで本論文では，第 2 章において，小型情報端末上での表示方式の改善を重視した検索支援方式について検討する．ここでは，Web ページの表形式情報は実世界において時間変化する重要な情報を反映してコンパクトに集約したものであることが多いことに着目し，「小型情報端末向け時系列 Web 情報自動抽出・提示手法」を提案する．提案手法では，表形式情報に適用先を限定することで，高い実用性を実現する．

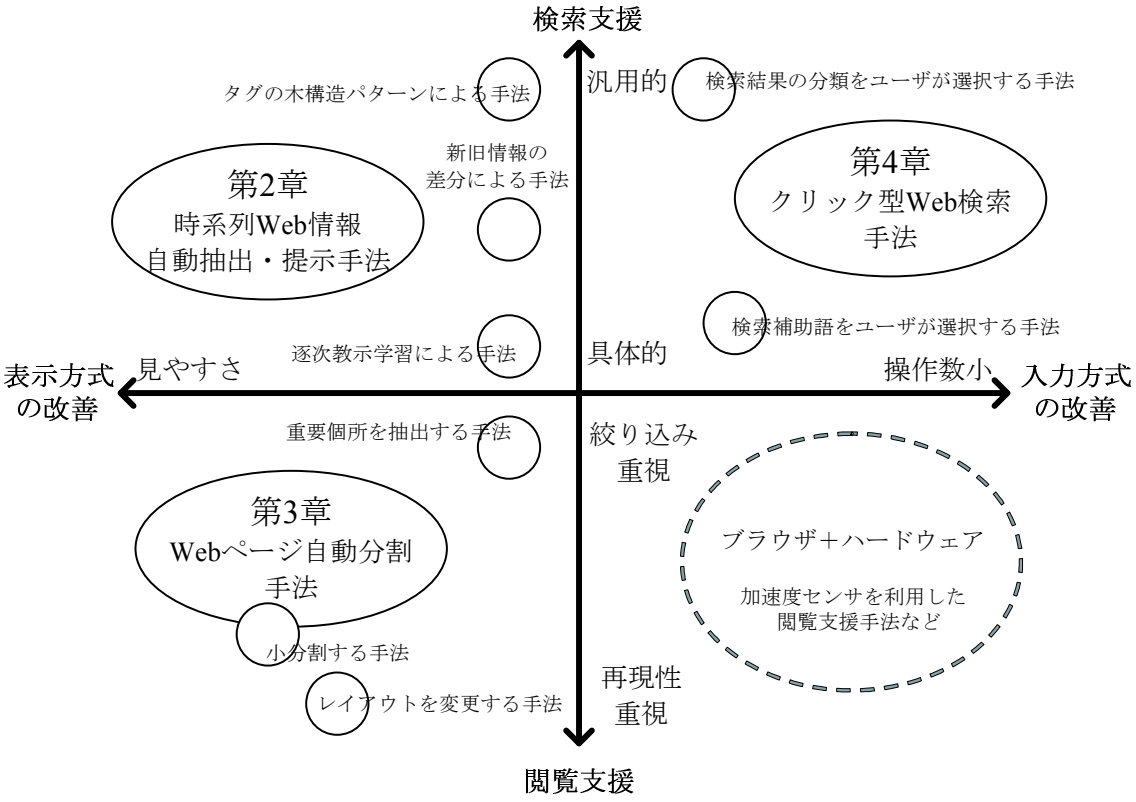


図 1.1: 小型情報端末向け情報収集支援関連の既存技術と本研究項目の位置づけ

次に，左下の「表示方式の改善による閲覧支援技術」の領域においては，既存研究として，再現性を重視する順に，レイアウトを小型情報端末向けに変換する方式 [8, 9, 11, 51, 56]，Web ページを小分割して再構築する方式 [12, 48, 49]，重要個所を抽出する方式 [1] がある．しかしながら，これらの方式は，小型情報端末においてユーザが閲覧したい情報に素早くアクセスできるようにするための表示方法に関する検討は行っていない．そこで本論文では，第 3 章において，小型情報端末上で見やすい表示方法を重視した閲覧支援方式について検討する．ここでは，Web ページを小分割して再構築する方式をベースに，より分割精度の高い,「小型情報端末向け Web ページ自動分割手法」を提案する．提案手法では，特に小さなサイズの画面での一覧性を重視する．

また，右上の「入力方式の改善による検索支援技術」の領域においては，既存研究として，適用先の汎用性の高い順に，検索結果の自動分類結果をユーザが選択しながら絞り込んでいく検索方式 [18, 43]，システムが提示した検索補助語をユーザが選択して絞り込む

検索方式 [36] がある．しかしながら，これらの方式は，小型情報端末では素早くかつ正確な操作が困難であるという問題点に着目した検討は行っていない．そこで本論文では，第 4 章において，検索操作でのユーザの操作を最小限にすることを重視した検索方式について検討する．ここでは，検索操作の契機の 1 つである Web ページ閲覧中に気になった単語に関する検索を行う場合に着目し,「小型情報端末向けクリック型 Web 検索手法」を提案する．提案手法では，最小で 1 回のクリック操作により適切な検索結果が得られることを実現することを目的とする．

最後に，右下の「入力方式の改善による閲覧支援技術」の領域においては，既存研究として，加速度センサなどのセンサと Web ブラウザを協調させることで，煩雑で面倒なスクロール操作から解放するなどの方式 [55] が検討されている．また，指紋読み取りセンサを利用して Web ブラウザのスクロール操作を支援する入力デバイス [34] を搭載した携帯電話端末が市販されている．しかしながら，他の 3 つの領域とは研究の方針が大きく異なることから，本論文では，Web 情報を分析することによる課題の解決方針に絞り，入力デバイスの高度化による解決方針については扱わないこととする．ここで，入力デバイスの操作性が評価結果に影響することを避けるため，汎用的なリモコン型 UI を入力デバイスとして想定する．

1.2.2 小型情報端末向け時系列 Web 情報自動抽出・提示手法

1 つの情報を追跡し，時系列でまとめる応用について検討する．例えば株価の時系列情報については専用サイトがあるが，任意の情報について時系列でまとめるサイトを個々に作成することは現実的ではない．ここでは，ユーザの要求時にオンデマンドで対応するシステムの実現を目的とする．

Web ページ中の個々の情報は値だけではなく表示位置も変化する場合があり，さらにその更新頻度が高いという特徴を持つ．そのためユーザが Web ページから情報抽出を行うには，ブラウジングや検索，抽出，集約といった作業を継続的にかつ頻繁に行う必要があり，作業コストがかかるという問題がある．特に小型情報端末では非常に煩雑な作業となる．よってこれらの作業をユーザの代わりに行うことで，ユーザの作業コストを削減する情報集約エージェントシステムの有効性は高い．このシステムの機能要件は，ユーザの作業を最小限にすること，表示位置が変化する情報を追跡して収集すること，さらに収集した情報を端的に表す適切なキーワードを付与してユーザに分かりやすい集約表示を軽量な処

理で実現することである．しかしながら，既存の情報集約エージェントシステムでは適切なキーワードを抽出する機能はなく，また抽出するためにはユーザの作業を伴うため，機能要件を満たさない．そこで，連想性という概念を導入し，ユーザの作業コストを最小限にしながら適切なキーワードを抽出することにより，全ての要件を満たす情報集約方式を提案する．

1.2.3 小型情報端末向け Web ページ自動分割手法

PC 向けの Web ページでは，携帯電話向けやカーナビゲーションシステム向けなどの小型情報端末向けの Web ページと比較して情報量が多い．ここで，小型情報端末ではあくまで臨時的な閲覧目的が多いため，表示面積の大きな PC 向けの Web ページのすべてを見る必要があるわけではなく，必要な一部の情報のみでよい場合が多い．そのため，本研究では，小型情報端末を利用している場合において，PC 向けの Web ページの中から必要な情報を簡単に発見して閲覧するためのシステムの実現を目的とする．

小型情報端末で Web ページの閲覧をしたいという要求は高まっているが，多くの Web ページは PC 向けに作成されているため，小型情報端末の画面サイズに合う情報を得ることは難しい．したがって，PC 向けの Web ページを小型情報端末向けに再構築することが必要となる．既存の方式には，HTML(Hyper Text Markup Language) ソースのタグ構造を利用する手法があるが，HTML ソースは不完全なタグ構造を許容するため，構造解析でエラーとなる可能性が高い．そこで，不完全なタグ構造に対応できるように，HTML ソースのタグ構造に加えてタグの深さ情報を利用して，Web ページを小型情報端末の画面に合う小さな Web ページに自動的に分割する，ロバスト性の高い Web ページ再構築手法を提案する．

1.2.4 小型情報端末向けクリック型 Web 検索手法

携帯電話を筆頭に小型情報端末を用いた Web 閲覧環境が整いつつある．ただし，小型情報端末は PC のようなキーボードを持たないため文字入力が面倒であり，検索を容易に行えない．Web 閲覧には検索が必須であるため，検索操作を容易にするシステムの実現を目的とする．

小型情報端末は，キーボードを搭載せず，数字ボタンや十字キーなどの限られたリモコン型 UI のみを備えているものが多い．そのため，Web ページの閲覧は容易に行えるもの

の，Web ページの検索を行う場面では，試行錯誤しながら検索語入力を行い検索結果を絞込むという PC と同等の検索手法は困難であり，小型情報端末向けの使い易い Web 検索方法が求められている．そこで，閲覧中の Web ページに表示されている語の中からユーザが検索したい語 (注目語) を指定すると，システムが自動的に最適なページを提示するクリック型の Web 検索手法を提案する．

1.3 本論文の構成

本論文の構成は次の通りである．まず，第 2 章で，小型情報端末を利用して Web 上の情報を定期的に収集したい場合における，ユーザの作業コストを削減するための，時系列 Web 情報自動抽出・提示手法について述べ，第 3 章において，小型情報端末を利用して PC 向けの Web ページを閲覧する際に，ユーザが目的とする情報に簡単に閲覧するための，Web ページ自動分割手法について述べる．次に，第 4 章において，小型情報端末を利用して検索する際に簡単に検索語の入力をできるようにするための，クリック型 Web 検索手法について述べる．最後に第 5 章において，本論文のまとめと今後の検討課題を述べる．

なお，第 2 章は文献 [24, 28] で公表した結果に基づき論述する．第 3 章は文献 [25, 26, 27, 29, 30, 31] で公表した結果に基づき論述する．第 4 章は文献 [32, 33] で公表した結果に基づき論述する．

第2章

小型情報端末向け時系列Web情報自動抽出・提示手法

2.1 まえがき

近年，Webで提供される情報は爆発的に増加しており，ハイパーリンクをたどるような単純な情報探索では，ユーザが必要な情報を発見することは困難な状況にある．そのため，多くのユーザは日常的に検索サイトを利用して情報検索を行い，多数のWebページから必要な情報を収集している．Webページで提供される情報の特徴として，紙媒体の新聞や雑誌等よりも情報がリアルタイムに更新されることが挙げられ，最新情報を得たいユーザにとっては非常に便利な情報源となりうる．しかしながら，それゆえにユーザが最新情報を得るためには，頻繁にWebページを訪問して情報収集を行う必要が生じる．このような継続的な情報収集はユーザの時間や労力等の作業コストがかかるタスクであり，小型情報端末では特に負担となる．よって，この作業コストを削減するためにシステムが支援することが必要であり，ユーザの代わりにユーザが必要とする情報を自律的に収集して集約表示する，Web情報自動抽出システムの構築が重要である．

ユーザが注目する情報は，週末の天気予報や特定銘柄の株価情報のように表形式でまとめられた情報 (以下，表形式情報と呼ぶ) であることが多い [65]．一方，Webページの作成者は，ハイパーリンクを利用したWebページの階層化による情報の分類や，テーブルタグを利用した情報の表形式による表示等の工夫により，ユーザが閲覧しやすいように整理した形で情報提供を行い，より良い情報源 [17, 35] になるように努力をしている．例えば図

2.1 に示すような天気予報ページの作成者は，地域ごとにハイパーリンクで分類し，1 週間の天気や気温等を整理して 1 つの表形式で表示している．また，価格比較サイトの作成者は製品カテゴリやメーカ名ごとに分類し，商品の販売価格を整理して表形式で比較表示している．このように Web ページの表形式情報は，実世界において時間変化する重要な情報を反映して，見やすくコンパクトに集約したものであることが多く，Web ページ上の情報の中でも特に重要な情報が含まれているといえる．

ただし表形式情報は，更新される度に個々の情報の文字列自体が変化するだけでなく，その表示位置も変化する特徴を持つ．そのため，固定的な位置にある情報を抽出するような単

図 2.1: 天気予報のサイト例

純な手法では，ユーザが指定した情報を継続的に抽出することには対応できない．文字列と表示位置が変化する情報を抽出するためには，抽出対象となる情報の指定のほかに，情報を同定するためのいくつかの追加情報が必要となる．しかしながら，小型情報端末を操作端末とした場合，画面の表示領域が狭く，表形式情報を概観できない場合も生じる．この場合，必要な情報を探すためのスクロール操作などが必要となり，複数の情報をシステム側に入力することはユーザの作業コストが高い．さらに，収集した情報を小型情報端末上で整理して集約表示する編集作業の作業コストが同様に高いため，システムは自動的に収集した情報を整理し，集約して提示する必要がある．

そのため，求められる Web 情報自動抽出システムは，ユーザの作業コストを最小限にしながら，表形式情報からユーザが指定した一部分の情報を自律的にトラッキングして収集し，収集した情報を表すキーワードを付与するなど，見やすい集約表示を実現できなければならない．しかしながら既存の方式は，新規の情報を収集対象とするために学習データを与えることや収集結果の情報に正誤の評価を与えること等，システムに対するユーザの作業コストが継続的にかかる課題や，表形式情報の固定的な位置にある情報をラベルにしてシステムが情報のトラッキングを行うため，適用可能な表形式情報が限定される課題がある．

そこで本章では，Web ページの表形式情報からユーザが指定した情報をトラッキングする Web 情報自動抽出システムにおいて,「連想性」という概念を導入して適切な情報のラベルを抽出することにより，ユーザの作業コストを最小限にしながら，情報のトラッキングとユーザに分かりやすい集約表示を実現する手法を提案する．

以下，2.2 節では，Web 情報自動抽出システムに関連する既存研究について述べる．2.3 節では，小型情報端末のための Web 情報自動抽出システムの機能要件を説明する．2.4 節では，提案方式であるオブジェクトに基づく自動抽出手法を提案する．2.5 節では，提案方式の性能評価を行い，最後に，2.6 節で本章のまとめを述べる．

2.2 関連研究

本節では，関連する既存の Web ページの情報収集方式を利用した Web 情報自動抽出システムについて，概要と問題点を述べる．これらの機能の比較表を表 2.1 に示す．ユーザが指定した情報を対話的分類学習を利用して自動抽出する方式 [65] があるが，ユーザがシステムに教示してルールを学習させる作業コストがかかることや，移動する情報の表示位置

を特定するために表の最上位行と最左列にあるセルの情報を項目名であると仮定しているため，それ以外の表示形式には適用できないという問題がある．新旧の Web ページの差分を算出して変化した部分を検出する方式 [52] があるが，検出する範囲をユーザが指定できないため，検出結果がユーザの見たい部分であるとは限らないという問題がある．収集対象の情報を囲む HTML ソースのタグパターンを木構造を解析して推定し，それをオブジェクトラベルとする方式 [13, 14, 21, 42, 61] があるが，パターン生成のための学習データの入力が必要であり作業コストがかかることや，ページ内に同じタグパターンの情報が複数存在する場合は収集対象を特定できないという問題がある．また，既存方式の組合せや機能分散等の応用により，Web からの情報抽出機能を容易に生成可能とするためのシステム構築方式 [2, 3, 41, 54] があるが，情報の抽出機能は既存の方式に依存しており，根本的な問題の解決には至っていない．また，集約表示を実現する場合に，表題などの属性情報を表形式情報から自動抽出しなければならないが，ほとんどの既存の方式においては検討されていない．いくつか検討事例はあるが，表形式情報の最上位行と最左列の文字を機械的に取得する程度の検討しかされていない．

以上の検討より，既存方式は，情報の表示位置が変化しない表形式情報，あるいは固定位置の情報を抽出する要求については対応することができる．本研究では，情報の表示位置が変化し，かつ抽出対象が移動するような表形式情報からの自動情報抽出を実現する手法について議論する．

2.3 Web 情報自動抽出システムの機能要件

2.3.1 システム要件

表形式情報の特徴として，表中の個々の情報が時間の経過に伴い列方向または行方向に移動することが挙げられる．例えば週間天気予報では，ある特定日の情報は値を変えながら毎日左に 1 つずつ規則的に移動し，また株価の売買高等のランキング情報では，各情報は一定時間毎に列方向に不規則に移動する．さらに，Web ページの情報は更新の頻度が高い特徴を持つ．紙メディアである新聞や雑誌の情報更新の頻度は半日から 1 週間程度であるのに対し，Web ページの情報はリアルタイムに更新されるものが多い．更新が多い場合は数時間で古い情報は表示されなくなり，ユーザは見ることができなくなってしまうこともある．そのため，ユーザが最新情報を漏れなく得るためには頻繁に Web ページにアクセ

表 2.1: 既存の Web 情報自動抽出システムの比較

	値が変化し表示位置が移動する情報への対応	分かりやすい集約表示	簡便なユーザ入力
(A) 対話的分類学習を利用して情報抽出を行う方式 [65]	△ 可能だが，表の 0 行 0 列で情報の表示位置を識別できない場合は適応できない	△ GUI 上に抽出した文字列を列挙するのみ	× URL と収集対象の指定に加え，抽出ルール生成のためのユーザからの教示による学習が必要
(B) 新旧の Web ページの差分を利用して情報抽出を行う方式 [52]	△ 可能だが，変化がない情報は抽出できない	× 検討なし	◯ URL の指定のみ
(C) タグの木構造パターンを利用して情報抽出を行う方式 [13, 14, 21, 42, 61]	× 表形式情報では同じタグパターンがページ内に複数存在するため特定できない	× 検討なし	× URL と抽出するタグパターンの指定に加え，パターン生成のための学習データの入力が必要
(D) 情報抽出システムを容易に生成可能とするためのシステム構築方式 [2, 3, 41, 54]	△ または× アーキテクチャに適用する方式 (A〜C) に依存する	× 検討なし	△ URL と収集対象の指定に加え，各情報の表示形式に対応するための学習データの入力が必要だが，最小限になるように工夫している

スして情報収集を行わなければならない．このタスクは時間や労力等の作業コストを要することが課題となる．

Web ページの表形式情報を対象とした Web 情報自動抽出システムとは，表形式情報から継続的に情報収集したいユーザの作業コストを削減する目的で，Web 情報自動抽出システムが自律的かつ継続的にユーザが指定した情報を収集し，集約表示によりユーザに結果を提示するシステムとする．システムのアプリケーション例を図 2.2 に示す．ユーザがある日の天気予報の変化を収集したい場合 (ここでは 8 月 22 日 (日)) は，週間天気予報の表形式情報の天気部分をクリックして指定する．すると Web 情報自動抽出システムは情報が更新される度にその日の天気情報を収集して時系列で集約し，結果をユーザに提示する．図 2.2 では 8 月 15 日から 6 時間おきに収集した例を示している．提示手段として電子メールや Web サーバを利用することで，携帯電話等の小型携帯端末でも集約結果を閲覧可能となる．

このようなシステムを実現するために必要な 3 つの機能要件を以下に示す．

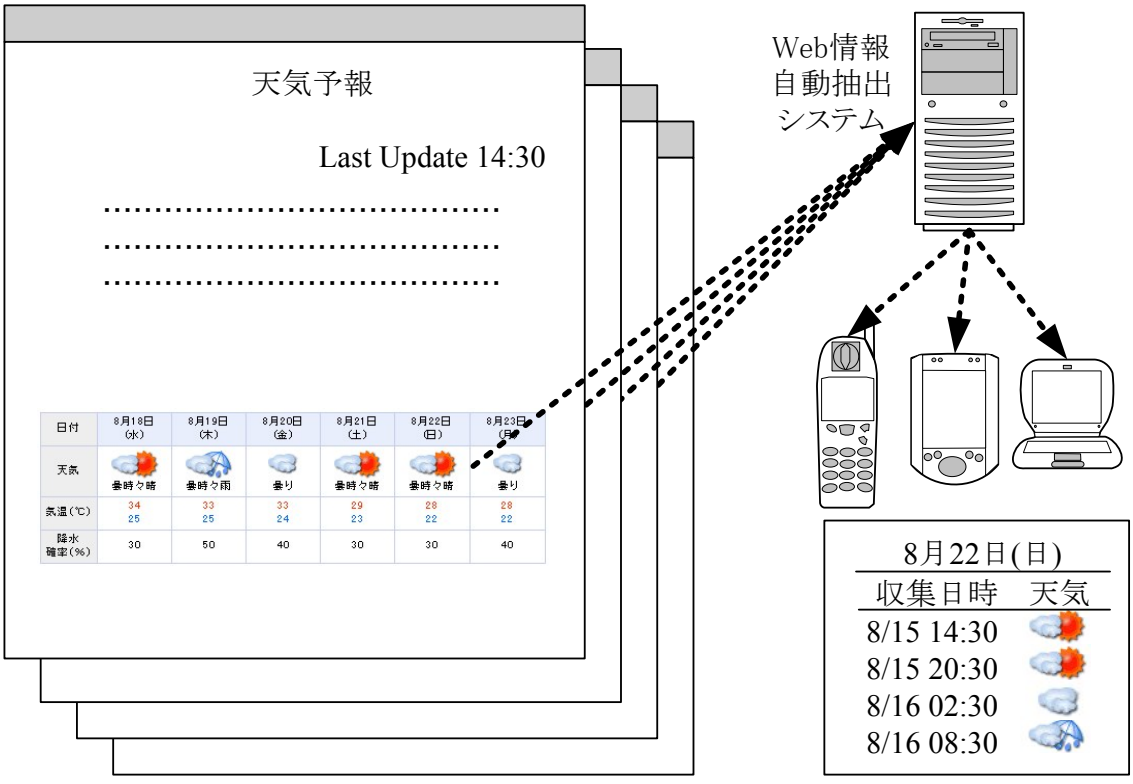


図 2.2: アプリケーション例

(機能要件 1) 時間の経過と共に値と表示位置が変化する情報のトラッキングができること

表形式情報の場合は時間の経過と共に値や文字列が変化するだけでなく，情報の表示位置が変化する場合があるため，Web 情報自動抽出システムはこのような情報をトラッキングして抽出しなければならない.

(機能要件 2) ユーザに分かりやすい集約表示ができること

複数の情報を同時に集約する場合には，それぞれの集約結果が何の結果であるかが分かりやすく示されていることが重要である．システムは収集した情報を収集日時順に並べて表示するだけでなく，項目名や標題等を付与して集約表示する機能が必要である.

例えば図 2.3 において，ユーザが上から 5 行目にある「ページデザイン (株)」の順位 (4)，取引値 (219,000)，前日比 (+30,000) の 3 つのセルを指定すると，システムは定期的に (ここでは 24 時間ごとに) 同一の情報を収集して図 2.4 に示すような時系列の集約表示を行う．ここで「ページデザイン (株)」は標題であり,「収集日時」はシステムが収集した日時,「順位」「取引値」「前日比」はユーザが指定した情報の項目名である．図 2.4 の集約表示を実現するためには，システムは図 2.3 の「名称」列のように行を特定可能，かつ意味を容易に理解可能な標題 (以下，連想性の高い標題と呼ぶ) となるセルを抽出しなければならない．さらに図 2.3 の 1 行目のような情報の項目名をそれぞれ表形式情報から自動的に抽出しなければならない．ただし情報の項目名については，表のセル間の類似度の差を利用して項目名と情報部分を自動認識して分離する方式 [51] を応用することにより自動抽出が可能であると考えられるため，連想性の高い標題を抽出する方式について新たに検討する必要がある.

(機能要件 3) ユーザによる簡易な収集対象の指定ができ，かつユーザがシステムに与える情報量を削減すること

Web ページ全体や表形式情報全体あるいはあらかじめシステムで決められた情報を収集対象とするのではなく，ユーザが指定した情報のみを収集対象とする仕組みが必要である．また，本システムはユーザの作業コストを削減することが目的であるため，ユーザがシステムに与える情報は最小限にしなければならない．よってユーザが Web 情報自動抽出システムに対して入力する情報は以下の 2 つのみとする.

・収集対象の情報を掲載している URL

・収集対象となる表形式情報の 1 つのセル

順位	コード	市場	名称	取引値		前日比		出来高
1	2406	JASDAQ	㈱アルデンテ	8/16 14:26	**635,000**	+100,000	+18.69%	2,167
2	2404	マザーズ	㈱建築計画	8/16 14:28	**1,330,000**	+200,000	+17.70%	2,980
3	1518	東証1部	小電力産業㈱	8/16 15:00	**195**	+28	+16.77%	88,229,000
4	4794	マザーズ	ベージデザイン㈱	8/16 09:54	**219,000**	+30,000	+15.87%	674
5	4797	マザーズ	㈱アイビー	8/16 14:59	**672,000**	+82,000	+13.90%	6,143

図 2.3: 株式ランキング情報表示の例 (表中の名称は実在のものではない)

ページデザイン(株)			
収集日時	順位	取引値	前日比
2004/8/16 10:10	4	219,000	+30,000
2004/8/17 10:10	30	110,000	-109,000
2004/8/18 10:10	2	908,000	+798,000

図 2.4: 集約表示の例

2.3.2 Web 情報自動抽出システム概要

2.3.1 項で示した 3 つの機能要件を満足する Web 情報自動抽出システムの処理シーケンスを図 2.5 に示し，以下に各処理の概要を述べる．

(1) URL を送信 (Send “URL”)

ユーザが，Web 情報自動抽出システムに対して，収集対象の情報が掲載されている Web ページの URL を入力する．

(2) リクエスト送信 (GET “URL”)

Web 情報自動抽出システムが，Web サイトに対して，URL に対応する HTML ソースを要求する.

(3) HTML 応答

Web サイトが，Web 情報自動抽出システムに対して，HTML ソースを返す.

(4) HTML に GUI を挿入

Web 情報自動抽出システムが，HTML ソースから表を抜き出し，情報選択用のボタン等を挿入して設定用 GUI を生成する.

(5) 設定用 GUI 応答

Web 情報自動抽出システムが，(1) の応答として (4) で生成した設定用 GUI の Web ページを返す.

(6) 収集対象を指定

ユーザは，設定用 GUI を利用して収集対象を指定する.

(7) 収集対象の初期状態を蓄積

Web 情報自動抽出システムは，ユーザが指定した収集対象の初期値と表全体を蓄積する.

(8) リクエスト送信 (GET “URL”)

Web 情報自動抽出システムは，定期的に URL に対応する HTML ソースを Web サイトに要求する.

(9) HTML 応答

Web サイトが，Web 情報自動抽出システムに対して，HTML ソースを返す.

(10) 表の解析・情報蓄積・集約表示

Web 情報自動抽出システムは，蓄積した最新の表形式情報と比較して変化がなければ (9) の HTML ソースを破棄する．変化があれば表の解析処理を行い，収集対象の情報を抽出して結果の集約表示を作成する．その後，処理 (8) に戻る.

(11) 集約結果要求

ユーザは，Web 情報自動抽出システムに対して，任意の時刻に集約結果を要求するメッセージを送信する.

(12) 集約結果応答

Web 情報自動抽出システムは，現時点までに情報集約した結果を返す.

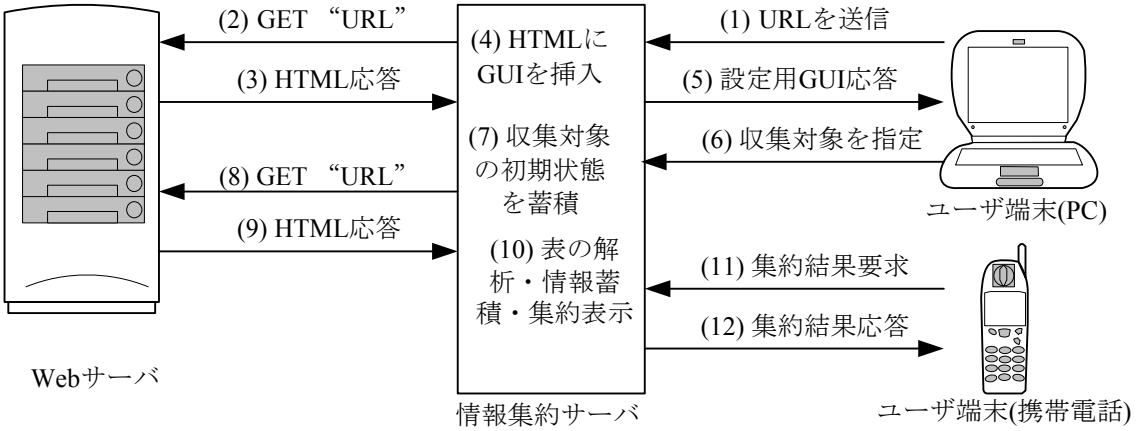


図 2.5: 処理シーケンス

2.4 オブジェクトに基づく Web 情報自動抽出手法の提案

機能要件 1 から 3 を満たす Web 情報収集システムを実現する手法として，表形式情報から情報の塊 (以下，オブジェクトと呼ぶ) を抽出し，オブジェクトを 1 単位としてユーザが指定した情報を自律的にトラッキングする仕組みを提案する．2.3 節で述べたように，表形式の各セルの情報は列方向や行方向に移動する性質を持つ．ただし，それらは完全に不規則に移動しているのではなく必ず関連する情報が 1 つのオブジェクトを成し，オブジェクト単位で移動している特徴が観測できる．また，そのオブジェクト内部における情報の移動は見られず，並び順の変化はない．つまり，表中の情報は列方向または行方向の何れかの方向にしか移動せず，かつその移動は必ずオブジェクト単位での移動となる．例えば図 2.1 ではオブジェクトは列方向の情報の塊であり各情報は行方向に移動し，図 2.3 ではオブジェクトは行方向の情報の塊であり各情報は列方向に移動する．よって，オブジェクトの移動方向を推定することで，ユーザが指定した情報をトラッキング可能である．具体的には図 2.3 においてユーザが「出来高」の列の「2,980」をクリックして指定すると，システムが行方向のオブジェクトであることを検出し，オブジェクトを特定可能な ID(Identification) として利用可能な「コード」の列の「2404」あるいは「名称」の列の「(株) 建築計画」を抽出する．この後システムはこれらのいずれかをキーとしてオブジェクトおよび指定した情報をトラッキングする．

提案システムでは，オブジェクトの認識とその ID を抽出することで，移動する情報を自

動的にトラッキング可能となり，機能要件 1 を満たす．また，ユーザは表形式情報の 1 つのセルを選択する操作のみで実現できることから，ユーザがシステムに入力する情報は URL と収集対象の 2 つのみであり，機能要件 3 を満たす．機能要件 2 については，連想性という新たな概念を定量化することにより，連想性の高い標題の抽出を可能にする．なお，抽出した表題がオブジェクトを一意に識別する ID としての性質を持つことから，本研究では，オブジェクトを特定可能，かつ連想性の高い表題 (以下，オブジェクトラベルと呼ぶ) を抽出することを検討する．なお，図 2.5 の処理 (10) は情報が変化するまで実行されないため，ユーザが最初に情報を指定した時点 (時刻 t_1 とする) の表形式情報 (Table_{t_1}) に対し，情報が変化した時点 (時刻 t_2 とする) の表形式情報 (Table_{t_2}) が情報集約サーバ上に蓄積された時点で，システムは処理 (10) を実行する．

以下では，表形式情報におけるオブジェクトの方向識別手法，ならびにオブジェクトを識別するための連想性の高いオブジェクトラベルの抽出手法の詳細を述べる．

2.4.1 オブジェクトの方向識別

図 2.6 に示すようにオブジェクトの方向に対する垂直の方向には，全く同じあるいは似たような文字列が並ぶことから，表の全ての行方向と列方向でセル間の類似度を算出し，類似度が高い方向に対して直交した方向がオブジェクトの方向であると判定することができる．類似度の算出方法として，ベクトル空間法により個々のセルの文字列の類似度を算出する方式 [51] がある．しかしながら，計算量が膨大であるという問題があるため，本研究では情報が移動する性質を利用した軽量な方式を採用する．具体的には移動前と移動後の表を比較して同一列または同一行にある完全一致する文字列を持つセル数をカウントし，カウントの総計が多い方向を移動した方向であると推定して，その方向に対して直交した方向をオブジェクトの方向とする．このオブジェクトの方向識別アルゴリズムを以下に示す．

(1) GUI を利用して，ユーザが時刻 t_1 における Web ページの表形式情報 (Table_{t_1}) の 1 つのセルを収集対象として指定する．
(2) Table_{t_1} が更新された時 (時刻 t_2) の表形式情報 (Table_{t_2}) を取得する．
(3) Table_{t_1} と Table_{t_2} から 2 つの Count テーブルを生成する．ここで，$C1_{ij}$ は Table_{t_1} の i, j 番目のセルのテキストを表し，$\overline{C2}_{i\cdot}, \overline{C2}_{\cdot j}$ は Table_{t_2} の i 行目のセルベクトルおよび j 列目のセルベクトルをそれぞれ表す．また，$duplicity(\overline{C}_{i\cdot}, C_{ij}) = \sum_n \mathrm{if}(C_{in} == C_{ij})$ とし，$\mathrm{if}(x)$ は x が TRUE のとき 1，x が FALSE のとき 0 を返す関数とする．

行方向の Count テーブル (RCT):

$$\{RCT_{ij}\} = \left\{ duplicity(\overline{C2}_{i\cdot}, C1_{ij}) \right\} \tag{2.1}$$

列方向の Count テーブル (CCT):

$$\{CCT_{ij}\} = \left\{ duplicity(\overline{C2}_{\cdot j}, C1_{ij}) \right\} \tag{2.2}$$

(4) RCT_{ij} および CCT_{ij} を利用し，オブジェクトの方向を下記の条件に従い判定する．下記以外の場合は，セル間の類似度を算出して決定する既存手法等を利用する [51].

行方向と判定:

$$\sum_{i,j} RCT_{ij} < \sum_{i,j} CCT_{ij} \tag{2.3}$$

順位	コード	市場	名称	取引値		前日比		出来高
1	2406	JASDAQ	(株)アルデンテ	8/16 14:26	**635,000**	+100,000	+18.69%	2,167
2	2404	マザーズ	(株)建築計画	8/16 14:28	**1,330,000**	+200,000	+17.70%	2,980
3	1518	東証1部	小電力産業(株)	8/16 15:00	**195**	+28	+16.77%	88,229,000
4	4794	マザーズ	ベーシデザイン(株)	8/16 09:54	**219,000**	+30,000	+15.87%	674
5	4797	マザーズ	(株)アイビー	8/16 14:59	**672,000**	+82,000	+13.90%	6,143

オブジェクト　オブジェクトラベル候補

図 2.6: オブジェクトとオブジェクトラベル候補

列方向と判定:

$$\sum_{i,j} RCT_{ij} > \sum_{i,j} CCT_{ij} \tag{2.4}$$

2.4.2 オブジェクト識別のための連想性の高いオブジェクトラベルの抽出

個々のオブジェクトを識別するためには，オブジェクトを一意に識別可能なオブジェクトラベルが必要である．オブジェクトラベルは各オブジェクトに必ず 1 つ存在し，その集合はいずれかの列または行となる．例えば図 2.6 ではオブジェクトラベルの候補はいずれかの列になることが分かる．オブジェクトを一意に識別できなければならないことが要件であることから，時刻 t_1 における表形式情報 (Table_{t_1}) および時刻 t_2 における表形式情報 (Table_{t_2}) において，同一列に全く同じ文字列を含む列は除外する．例えば図 2.6 では「市場」「取引値 (左)」の列が除外される．残りは「順位」「コード」「名称」「取引値 (右)」「前日比 (左，右)」「出来高」の列であり，これらがオブジェクトラベルの候補となる．ここで，本質的なオブジェクトラベルの列は「コード」か「名称」が適切であると考えられ，その他の列は偶然同じ数値や文字列が存在しなかったために候補に上がっているだけである．さらに人が判定する場合は，認識しやすい「名称」の列をオブジェクトラベルとして選択する．そのため，検索システムにおいても，集約表示の標題にはこのラベルが選ばれることが望ましい．しかしながら機械的に判定する場合は，一意にオブジェクトを識別可能であるという条件以外には明確な選択基準がないため，これ以上オブジェクトラベルの候補を絞ることはできない．よって新たな選択基準が必要である．例えば，候補の中からランダムに選択することも考えられるが，候補の全てが正しいオブジェクトラベルであるとは限らない．

そこで，正しいオブジェクトラベルを機械的に選択可能とするため，人の判定基準に近い基準として「連想性」という概念を導入し，これを数値化することを検討する．ここで「連想性が高い」文字列とは人が理解しやすい文字列であり,「コード」の列のような数値や記号からなる ID ではなく,「名称」の列のように文字列を見るだけでその言葉が何を指すのかを連想できなければならない．

人が文字列の連想性が高いと判断する基準について検討すると，文字列自身が示す意味が明確であることが挙げられ，文書や話し言葉においてその文字列が利用される範囲が限定されていると解釈できる．連想性の度合いを数値化する方法として，連想性が高いと判断した文字列とその度合いを記述した辞書を人手で作成する方法が考えられるが，辞書の

更新等は人手の作業であり管理コストが継続的にかかる問題がある．そのため，人手で辞書を作成することなく自動的に最新の情報を参照可能とする仕組みが必要である．そこで，インターネット上の Web ページでその文字列が利用されている頻度を計測し，その頻度が最も低い文字列を連想性の高い文字列であるとする評価方法 (以下，連想性評価と呼ぶ) を提案する．具体的には，検索エンジンで各文字列の検索ヒット数を計測し，検索ヒット数が少ない文字列を連想性が高いと判断する．ここでは，利用される範囲が限定されている文字列とは，特定の分野においてのみ利用される可能性が高いと解釈する．このとき Web のようなあらゆる分野の文書が存在する世界を検索対象とした場合，利用範囲が限定されている文字列の出現頻度は，限定されていないものと比較して低いと考えることができる．

図 2.7 に連想性の高いオブジェクトラベルの抽出アルゴリズムを示す．図中の文字数制限のパラメタ p は定数とする．「連想性評価を行う」とは，検索エンジンを利用した検索ヒット数の抽出を行い，最も検索ヒット数が少ないオブジェクトラベルを選択する処理を表す．なお，連想性評価はオブジェクトラベル列 (または行) の候補が 2 列 (または行) 以上残った場合にのみ行うこととする．また，「1 ステップ戻る」とは，図 2.7 中の Step$k$ からこの処理に入った場合に Step$(k-1)$ の結果に戻す処理を表す．

以下に連想性評価の処理手順を示す．なお q, r は定数とする．

(1) 候補の行 (列) の全てのセルのテキストについて，検索ヒット数の抽出を行う．セル数が多い場合は最初の q セルのみに限定して行う．
(2) 抽出した検索ヒット数について，特異点を除くため最大値と最小値を取り除き，残りの平均値を評価値とする．
(3) 数値のみの場合は結果を r 倍して重みを付与する．

2.5 性能評価

2.5.1 連想性の高いオブジェクトラベル選択精度の評価

連想性評価を利用したオブジェクトラベルの選択方式の性能評価を行った．以下に測定条件と測定方法を示す．

(ア) 測定条件

(1) オブジェクトラベル抽出処理のアルゴリズムにおける 3 つのパラメタ (p, q, r) の設定値は，いくつかの表形式情報に対して予備実験を行った結果，最適であった $(p, q, r) =$

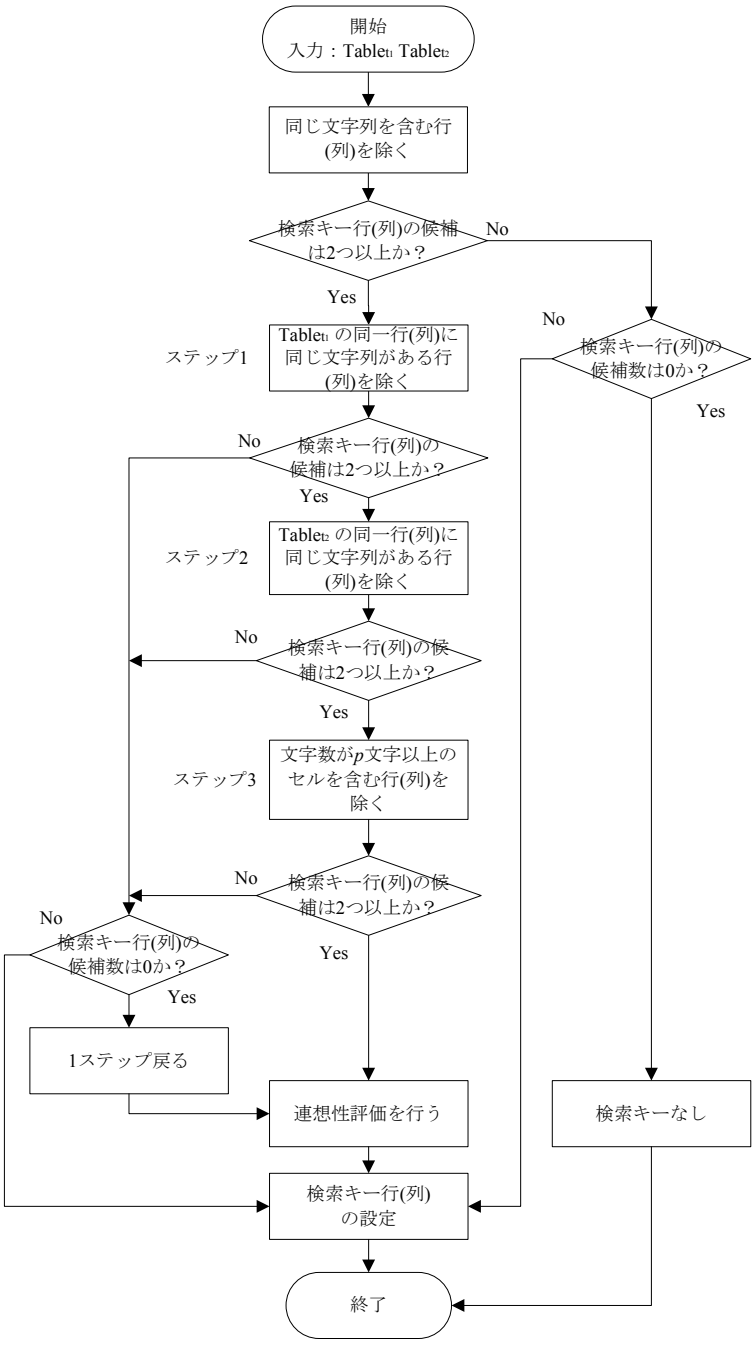


図 2.7: オブジェクトラベル抽出処理フロー

$(20, 10, 1000)$ とした.

(2) ユーザが指定する収集対象のセルは，表形式情報毎に 1 つのみとした.

(3) 連想性評価用の検索エンジンとして Google[22] を利用した.

(イ) 測定方法

(1) 評価対象の Web ページは，手作業で収集した 121 の Web ページとした．一部の Web ページの URL とその表形式情報の特徴を表 2.2 に示す.

表 2.2: 対象とした Web ページの URL と特徴 (一部)

	サイト名	URL	特徴
1	@映画生活	http://www.eigaseikatu.com/	上下に移動．順位，タイトル，その他数値情報
2	からしレンコンランキング	http://renkon.gfi-net.co.jp/	上下に移動．順位，タイトル＋短い解説，画像その他情報
3	春日井消防署リアルタイムデータ	http://www.city.kasugai.aichi.jp/	移動しない． 観測場所，日時，その他数値情報．最大 5 桁の数値を含む.
4	福岡とうもろこし先物	http://www.ojs-net.co.jp/	更新毎に情報が右に追加される．数値情報.
5	朝日新聞天気	http://weather.asahi.com/	更新毎に左に 1 つずつ移動．1 行目が日付 (オブジェクトラベル).
6	読売新聞天気	http://www.yomiuri.co.jp/	更新毎に左に 1 つずつ移動．2 行目が日付 (オブジェクトラベル).
7	懸賞ランキング	http://present.yahoo.co.jp/	上下に移動．順位，プレゼント名，その他数値情報.
8	yahoo 野球順位	http://sports.yahoo.co.jp/	上下に移動．順位，球団名，その他数値情報.
9	田中貴金属	http://www.tanaka.co.jp/	移動しない．コイン名称，サイズ，その他数値情報
10	yahoo 株価	http://biz.yahoo.co.jp/	上下に移動．順位，コード，市場，名称，その他数値情報.

(2) システムが正しいオブジェクトラベルを選択できたかどうかは人手で判定した.
(3) 結果について以下の 2 点で評価を行った.

(A) 連想性評価を行ったかどうか

オブジェクトラベルの候補が複数残ったため連想性評価が必要となった場合を「連想性評価利用」とし，オブジェクトラベルの候補が 1 つしか残らなかったため連想性評価を利用しなかった場合を「連想性評価不用」とした.

(B) 自動抽出した結果と人が判定した結果が一致するかどうか

システムが自動抽出したオブジェクトラベルと人が判定して抽出したオブジェクトラベルを比較し，結果が同じ場合を「最適な抽出」，異なる場合を「最適でない抽出」とした.

評価結果を表 2.3 に示す．全体の再現率は約 0.82(= 99/121) であった．また，連想性評価を行った対象は 31 例であり，これは表 2.3 の (A)「必要」の行の合計値が該当する．連想性評価を必要とする割合が約 0.26(= 31/121) であり，約 26%の表形式情報が連想性評価を必要としていたことになる．このうち，連想性の高いオブジェクトラベルの抽出に成功したのは 25 例であり，連想性評価の再現率は約 0.81(= 25/31) となった.

連想性の高いオブジェクトラベルの抽出に失敗した 22 件について，理由を以下にまとめる.

(1) オブジェクトの説明書き等の長文 (20 文字以上) を含むセルが正しいオブジェクトラベルであった場合に，他の列を選択した．(11 件)
(2) 複雑なセルの結合を行っており，オブジェクトを正しく抽出できなかった．(4 件)
(3) 長文同士の連想性評価となり，長い方のテキストを含むセルを選択した．(3 件)

表 2.3: 評価結果

		(B)		合計
		最適な抽出	最適でない抽出	
(A)	連想性評価利用	25	6	31
	連想性評価不用	74	16	90
合計		99	22	121

(4) オブジェクトラベルとなるべき列または行が画像のみである場合は連想性評価のためのテキストがないため，自動抽出できなかった．(2 件)
(5) セル内の文字列として表形式情報作成者が独自の略称を使用していた場合は検索ヒット数が小さい値となるため，これを選択した．(1 件)
(6) 識別子が存在しなかった．(1 件)

理由 (1) が 11 件で最も該当件数が多く，理由 (2)〜(6) への該当件数は比較的少数であった．理由 (1) が生じる原因は，2.4.2 項のアルゴリズムにおいて，長文テキストと判断した場合は他に候補があればオブジェクトラベルの候補から外しているためである．これは，長文のテキストをそのまま検索ワードとして検索エンジンに入力した場合，検索ヒット数が一般的に小さい値となり，常に最も連想性が高いと判定されてしまう問題があるために，これに対応する目的で導入している．逆に，この判断で正しいオブジェクトラベルを抽出できた場合は 38 件確認できており，再現率を上げることに対してはこのアルゴリズムは有効に働いているといえる．

連想性評価を利用しなかった場合においてオブジェクトラベルの選択に成功した例は，オブジェクトラベルとして利用可能な行または列が 1 通りしかなかった場合である．これについては人が選択しても結果は同じである．

また，表 2.2 の 9 番の例では連想性評価を利用せず，かつ失敗した．この Web ページでは硬貨の買取りレート情報を表形式情報として提供しており,「硬貨の種類」と「大きさ」の 2 つの列を組み合わせることで連想性の高いオブジェクトラベルとなる．本方式においては 1 列または 1 行のみで識別可能なオブジェクトラベルを抽出しているためこの種類の表形式には対応できていないが，隣接する複数行を組み合わせた場合についても対象とすることで対応可能となる．

2.5.2 処理時間の評価

Web 情報自動抽出システムのパフォーマンスを評価するために，システムの処理時間を測定した．システムの処理のうち最も時間を要する処理は，オブジェクトの方向判定処理と連想性の高いオブジェクトラベル抽出処理の 2 つであるため，主にこの 2 つの処理について評価を行った．測定条件と測定方法を以下に示す．

(ア) 測定条件

(1) PC1 台 (CPU:3.4GHz, メモリ:2GB) を利用した．また，2.4.2 項のアルゴリズムにおける 3 つのパラメタは 2.5.1 項の測定条件と同様に $(p, q, r) = (20, 10, 1000)$ とした．
(2) 対象とした Web ページの中で，行数と列数が最も多い表 2.2 の 10 番の表形式情報 (51 行 10 列，1 行目は項目名) を測定対象とした．通信性能や対象 Web サイトの処理性能の影響を避けるため，5 回分の変化した対象 Web ページの HTML ソースをシステムのローカルディスク上にあらかじめ蓄積しておき，Web 情報自動抽出システムはそれを参照することとした．1 分毎にファイルを更新し，これを 5 回繰り返した．
(3) オブジェクトの方向判定処理および連想性の高いオブジェクトラベル抽出処理は，一度確定すれば冗長な処理である．そのため最初の 2 回行った結果が全く同じであればオブジェクトの方向とオブジェクトラベルが確定したと判断し，それ以降はこの 2 つの処理を冗長として実行しないこととした．
(4) 連想性評価用の検索エンジンとして Google を利用した．

(イ) 測定方法

(1) システム全体の処理時間，オブジェクトの方向判定処理時間および連想性の高いオブジェクトラベル抽出処理時間をそれぞれ 10 回ずつ測定し，その平均値を算出して測定結果とした．これを対象の Web ページが更新される度に行った．
(2) 連想性の高いオブジェクトラベルの抽出に要する時間のほとんどは検索サイトとの通信時間であるため，システム全体の処理時間の結果はこの時間を除いた値とした．

結果を図 2.8 に示す．図 2.8 より更新回数が 1 回目および 2 回目のシステム全体の処理時間は 1000〜1200 ミリ秒程度であり，その約 40%の 500 ミリ秒程度がオブジェクトの抽出処理であった．測定対象の表形式情報では 2 列がオブジェクトラベルの候補として残っており，連想性の高いオブジェクトラベルの抽出処理は約 2300〜2800 ミリ秒程度であった．これにはインターネットを介した通信時間が含まれる．なお連想性評価においては，10 件ずつ並列に合計 20 件のクエリを検索エンジンに発行した．検索エンジンを利用した連想性評価を利用した場合はこの処理時間が加算されるため，全体の処理時間は 3300〜4000 ミリ秒となる．この処理については，検索方法に応じて大きく所要時間が異なり，辞書や検索システムが本システムのサーバ上にある場合は，より短時間で処理が完了すると思われる．また，3 回目以降はオブジェクトの抽出処理とオブジェクトラベルの抽出処理の 2 つの処理を冗長と判断してスキップし，定常状態としてそれら以外の処理のみを実行している．そ

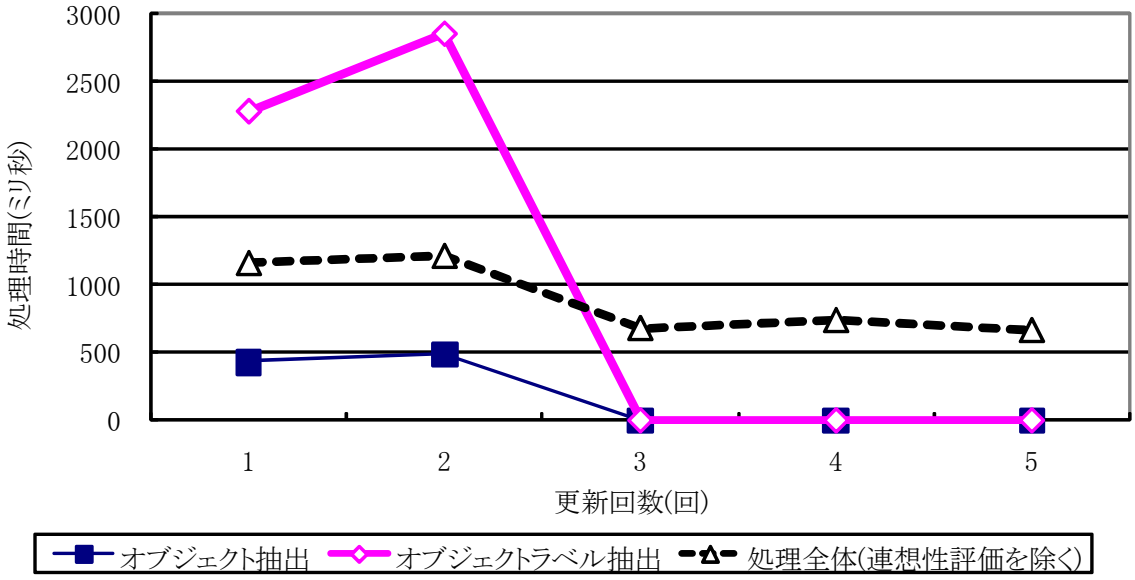


図 2.8: 処理時間測定結果

の結果，合計の処理時間が約 700 ミリ秒となり，処理時間を大幅に削減している．この処理時間はリアルタイムであるとはいえないが，測定対象の表形式情報の行数と列数は用意した Web ページの中で最も大きいことから，測定した処理時間は最大値に近い数値であると考えられること，および Web 情報自動抽出システムの利用形態はサーバ側でのバックグラウンド処理であることから問題のない処理時間であるといえる．

2.6 むすび

本章では，ユーザが Web から情報収集を行う場合のタスク軽減のため，表形式情報を対象とした Web 情報自動抽出システムについて検討した．表形式情報の特徴である表示位置が移動する情報への対応やユーザに分かりやすい集約表示等の機能要件を満たすため，表形式情報をオブジェクト化する方式と，オブジェクトを一意に識別可能な連想性の高いオブジェクトラベルを自動抽出する方式を提案した．評価システムを構築して連想性の高いオブジェクトラベル抽出機能の評価実験を行い，約 0.81 の再現率で連想性の高いオブジェクトラベルを抽出できることを示した．また，システムの処理時間の評価実験を行い，表形式情報に対して定常状態では最長約 700 ミリ秒程度で処理が完了する見通しが得られ，利

用形態がシステムによるバックグラウンド処理である場合は問題のない性能であることを示した.

実用化に向けた今後の課題として，オブジェクトラベルの抽出精度を向上することが明確な課題であるが，加えて，表形式情報の多種の表示パターンに対応することで，オブジェクトの抽出精度を向上することが挙げられる．2.5.1 項で述べたオブジェクトラベルの抽出に失敗した具体例のうち，(2) は，オブジェクトが 1 行または 1 列で表現されていない複雑な表形式情報であった例である．しかしながら，Web ブラウザ上では，抽出対象の情報とオブジェクトラベルとの対応関係をユーザが容易に理解できるように表示されており，本研究でまだ利用していない表形式情報の特徴がある．例えば，HTML ソースの `<table>` タグ構造の抽出に加えて，繰り返し構造や罫線の有無等の表形式情報の表現手法に基づくセル間の関連性を分析する手法が考えられる.

第3章

小型情報端末向けWebページ自動分割手法

3.1 まえがき

多くの小型情報端末はWebブラウザを搭載している．例えば，日本の携帯電話は2008年9月時点で約9000万台がWebブラウザ機能を搭載しており[60]，カーナビゲーションシステムの一部の機種はインターネットに接続する機能を持つ．このような小型情報端末の普及により，多くのユーザは鉄道の乗り換え案内や天気予報，最新ニュース等の様々な情報をいつでもどこでも得ることができる環境にある．しかしながら，小型情報端末の場合は小さな画面や自由度の低い入力デバイス等，ユーザインタフェースの制限がある．そのため，例えば携帯電話向けには小さい画面サイズに合わせた専用のWebサイトが存在するが，それらのWebページが持つ情報量は少なく調整される傾向にある．そこで，小型情報端末を利用して情報量が豊富なPC向けのWebページを閲覧する要求が高まっており，実際に，PC向けのWebページを表示可能なブラウザを搭載した携帯電話端末が市販されている．しかしながら，PC向けのWebページは，PC上で閲覧することを想定とした画面のレイアウトや情報量に調整されており，比較的大きな画面とキーボードやマウス等の自由度の高い入力インタフェースを利用しなければ容易に閲覧することはできない．携帯電話でPC向けのWebページを容易に閲覧するためには，Webページを小型情報端末向けに変換する必要がある．

この課題に対し，ユーザが関心を持つ情報はWebページの一部のみである場合も多いこ

とから [24, 65]，Web ページの情報をフィルタリングする等，ユーザが必要とする情報を容易に選択可能とすることで，表示領域の狭さと入力インタフェースの制限に基づく要因を回避する方針が有効であると考えられる．既存の手法として，HTML の文書構造を全体的に解析して得られる絶対的なタグの構造を利用し，Web ページを小分割する手法 [12, 49] が提案されている．しかしながら，HTML ソースにおけるタグの省略等を含む非正則な Web ページには適用できないという問題があった．そこで本章では，HTML ソースの部分的な解析により得られるタグの数や深さ等の相対的な階層構造を利用して算出した距離と，Web ページの大まかなレイアウト情報に基づき，Web ページを小分割する手法を提案する．本手法は，部分的かつ相対的なタグ構造を利用することから，原理的に HTML ソースの非正則な記述が解析の成否に影響しないため，非正則な Web ページにも対応できる．また評価実験を行い，本手法の有効性の検証を行う．なお，以降では携帯電話端末に絞って議論および評価を行う．

以下，3.2 節では，Web ページの自動分割に関連する既存研究と，小型情報端末のための自動分割システムの機能要件について述べる．3.3 節では，コンテンツ間距離とレイアウト情報に基づく Web ページ自動分割手法を提案する．3.4 節では，提案方式の分割精度の評価を行い，3.5 節では，提案方式を携帯電話上に実装してユーザビリティ評価を行う．最後に，3.6 節で本章のまとめを述べる．

3.2 関連研究

本節では，携帯電話を利用して PC 向けの Web ページを効率的に閲覧可能にする Web 閲覧システムについて，関連研究とその問題点を述べる．さらに，これらの問題を解決するための Web 閲覧システムの機能要件を示す．

3.2.1 関連研究とその課題

関連研究は大きく 4 通りに分類できる．以下にそれぞれの既存研究について述べる．

(1) Web ページのレイアウトを携帯電話向けに最適化する手法

携帯電話の狭い画面幅に合わせて，ブラウザが Web ページのレイアウトを縦長に変更する手法がある [56]．この手法は，PC を利用した閲覧に近い画面表示を実現することを特徴とする．しかしながら，ユーザが必要とする情報は Web ページの一部のみである状況も多

いことから [24, 65]，ユーザが見たい情報が下のほうに配置されていたときには長いスクロール操作が必要となる．そのため，その情報を探し出して閲覧するまでに手間がかかるという問題がある．また，表形式の情報を携帯電話向けにレイアウト変更する手法がある [11, 51]．この手法は，表の項目名を抽出して一行分ずつ見やすくレイアウトすることを特徴とする．しかしながら，これらの手法は，表形式でレイアウトされていない情報には応用が困難である．これらの手法の他にも，Web サイトのサイトマップのページを解析して携帯電話向けのメニューを自動生成する手法や [8, 38]，Web ページ中のリンクの重要度を利用者が判定し，重要度に基づくコンテンツのリストを自動生成する手法がある [9, 10]．これらの手法はサイトマップのページが必須であることや，選択したリンク先のページを閲覧するためには結局広い画面が必要となるという問題がある．

(2) Web ページを分割して再構築する手法

HTML ソースの階層構造を解析して Web ページを小分割し，写真とその解説文のようにレイアウト上で関連性が高いコンテンツの集合 (以下，コンテンツオブジェクトと呼ぶ) を生成する手法がある [12, 49]．この手法では，大量の情報を持つ Web ページの場合でもコンテンツオブジェクトごとに携帯電話の画面上に表示できるため，小さな画面でも容易に閲覧できる特徴がある．ここで，HTML ソースの構造を利用してコンテンツオブジェクトへの分割を行うため，正則な HTML ソースでなければならない．しかしながら，一般の Web ページは HTML ソースにおいて一部の終了タグが省略されていることや，HTML 標準には定義されていない不明なタグが挿入されている非正則な Web ページも多いため，適用できない Web ページが多いという問題がある．また，一つの Web ページを分割して複数の携帯電話の画面に割り当てて，協力して閲覧することを目的とした手法がある [48]．この手法は HTML ソースのタグ構造を重み付きの完全グラフに変換して分割することを特徴としているため，正しいグラフが生成できることが前提であり，非正則な HTML ソースには適用できないという問題がある．

(3) Web ページから重要部分のみを抽出する手法

ユーザのアクセス履歴等から嗜好情報を抽出し，単語の出現頻度をカウントしてユーザのプロファイルを作成することで，ユーザの嗜好に合わない部分を削除して小さな Web ページを再構成する手法がある [1]．この手法は，自動的に情報を取捨選択して Web ページを小さくできる特徴を持つが，再構成した Web ページが携帯電話で容易に閲覧可能な程度まで十分に小さくなるとは限らないという問題がある．

(4) Web ページの一部を拡大表示する手法

Web ページを携帯電話の画面に合うまで縮小したサムネイルを表示し，ユーザが閲覧したい部分を指定するとその部分を拡大表示する手法がある [7, 45, 63]．これらの手法は，Web ページ全体を概観することができるが，携帯電話の画面に合わせて縮小したサムネイル画像ではほとんどの文字が判読できないため，あらかじめ対象の Web ページのレイアウトに関する十分な知識がユーザになければならないという問題がある．

3.2.2 Web ページ自動分割システムの概要と機能要件

3.2.1 項で述べた関連研究の問題点を集約すると，携帯電話向けの Web コンテンツ閲覧システムの機能要件として，次の二点が挙げられる．

(機能要件 1) 1 ページの情報量を小さくすること

1 ページの情報量が多いとスクロールなどのユーザの操作が増えてしまい，閲覧が不便であるため，一度に表示する情報量を制限する必要がある．

(機能要件 2) あらゆる HTML ソースに対応できること

閲覧の対象となる Web ページは，タグの省略や未知のタグの挿入などを含む非正則な Web ページに対しても処理できる必要がある．

3.3 コンテンツ間距離とレイアウト情報に基づく Web ページ分割手法の提案

機能要件 1 および機能要件 2 を満たすため，`<table>` タグ等に基づくレイアウト情報に加えて，タグの深さと個数に基づいて算出するコンテンツ間距離を利用した，Web ページの分割手法を提案する．複数のコンテンツオブジェクトに分割する手法とすることで，機能要件 1 を満足する．さらに，タグの個数と相対的な深さに基づく分割手法を取り入れることで，タグの文法や種類に拠らない特徴を持つ．そのため必要なタグの省略や未知タグの挿入等の，非正則な HTML ソースに対する耐性が高く，機能要件 2 を満足する手法となる．

3.3.1 コンテンツ間距離を利用した Web ページ分割手法

コンテンツ間距離を利用した Web ページの分割手法は次の (A)〜(C) の三つのステップからなる．なお，本研究では「コンテンツ」を以下の (ア)〜(ウ) の三種類と定義し,「コンテンツオブジェクト」を関連するコンテンツの組と定義する．

(ア) HTML ソース中の <a> タグで指定されているアンカー

(イ) <img> タグで指定されている画像

(ウ) テキスト

(A) コンテンツの抽出

対象の Web ページの HTML ソースから三種類のコンテンツを抽出する．まず，<a> と で囲まれている文字列を抽出し，アンカーのコンテンツとする．次に，それ以外の <img> を画像のコンテンツとする．最後に，タグの属性値でない文字列について，<p> タグや <font> タグなどの文章構成や文字修飾以外のタグで区切られている単位でコンテンツとみなす．

(B) コンテンツ間距離の算出

隣接するコンテンツの間に挿入されているタグの数とその深さに基づき，コンテンツ間距離を算出する．ここでタグの深さとは，開始タグであれば直前のタグの深さに +1 し，終了タグがあれば −1 として部分的に累積して算出した値である．図 3.1 に概念図を示す．横軸 x はタグの順序番号を表し，HTML ソースの先頭にある <HTML> タグから数えた番目とする．縦軸 y はタグの深さとする．コンテンツ a とコンテンツ b の距離を $S_{(a,b)}$ と定義し，式 (3.1) に従い導出する．式 (3.1) において，x_a および x_b はそれぞれコンテンツ a およびコンテンツ b のタグの順序番号であり，y_a および y_b はそれぞれコンテンツ a およびコンテンツ b のタグの深さである．また，$f_{ab}(i)$ はコンテンツ a,b 間における $x = i$ のときの y の値を示す．なお，式中の $x_a, x_b, y_a, y_b, f_{ab}(i)$ は図 3.1 と対応している．

$$S_{(a,b)} = \max\left\{ \sum_{i=x_a}^{x_b} |\max\{y_a, y_b\} - f_{ab}(i)|, \sum_{i=x_a}^{x_b} |\min\{y_a, y_b\} - f_{ab}(i)| \right\} \tag{3.1}$$

(C) コンテンツオブジェクトへの分割

コンテンツ間距離の大小を比較して二分割を再帰的に繰り返し，Web ページを複数

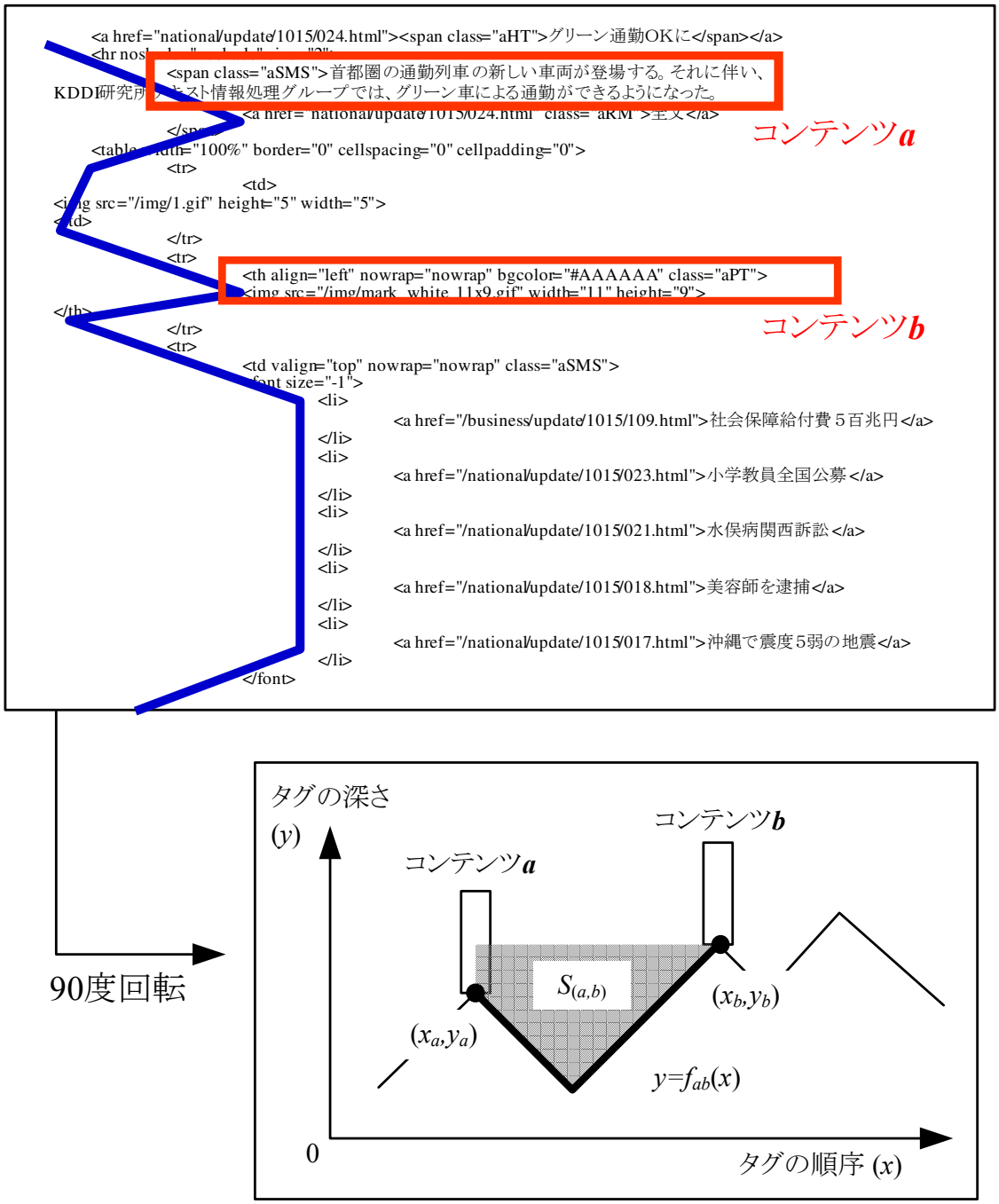


図 3.1: タグの深さとコンテンツ間距離

のコンテンツオブジェクトに分割する．手順を以下に述べる．ここで，$N_1 > N_2$ とする．

(1) Web ページ全体を一つのコンテンツオブジェクト ($ObjectID$ =root) とする．
(2) コンテンツオブジェクト内のコンテンツ間距離の最大値 ($S_{\max}$) が，コンテンツオブジェクト内のコンテンツ間距離の平均値 (S_{average}) の N_1 倍以上であれば，$S_{\max}$ の位置を分割点とする．
(3) (2) が真でない場合，S_{average} の N_2 倍以上かつ分割した場合のコンテンツ数の最小値が M 個以上であれば，$S_{\max}$ の位置を分割点とする．
(4) 分割した場合は分割結果の左のコンテンツオブジェクトに移動し (2) に戻る．そうでなければ (5) に進む．
(5) 左のコンテンツオブジェクトであった場合は，右のコンテンツオブジェクトに移動し，(2) に戻る．
(6) 右のコンテンツオブジェクトであり，かつ $ObjectID \neq (root)$ の場合は，親コンテンツオブジェクトに移動し，(5) に戻る．それ以外は (7) に進む．
(7) 終了．

ここで，Web ページを生成する市販ソフトを利用した場合と，HTML ソースをテキストエディタで作成する場合において，タグの複雑さは大きく異なる．つまり，最適となる分割の閾値 N_1, N_2 は Web ページごとに異なることから，特定の Web ページ対して最適な閾値を手動で設定できたとしても，すべての Web ページに対して設定するのは困難である．そのため，各々の Web ページに最適な閾値を動的に決定する手法が必要である．次項では，コンテンツ間距離の標準偏差に基づき，Web ページごとに分割の閾値 N_1 と N_2 を動的に決定して Web ページを分割する手法について述べる．

3.3.2 コンテンツ間距離の標準偏差に基づくパラメタ自動設定手法

まず予備実験を行い，N_1 と N_2 の自動設定の方針を定める．次にパラメタ自動設定手法の詳細を述べる．

(A) 予備実験

コンテンツ間距離 S の分布に応じた N_1 と N_2 の設定方法を検討するため予備実験を行う．まず，Web ページごとの N_1 と N_2 の最適値を求め，N_1 と N_2 の相関につい

て検証する．次に，N_1, N_2 とコンテンツ間距離とその統計値との関連について検証する．予備実験の詳細と結果を以下に述べる．

(1) N_1 と N_2 の相関について

Yahoo!カテゴリ [66] および Google ディレクトリから 106 のニュース提供ページと 104 の金融関連ページを任意に選択し，各 Web ページについて 3.3.1 項で記述した分割パラメタ N_1, N_2 を 1 から 16 の整数値の範囲で変化させてそれらの最適値を求めた．なお M は固定値 $M = 2$ とした．最適値は人の目で見て判断し，最も適切な位置でコンテンツオブジェクトに分割できた値とした．ニュース提供サイトと金融関連サイトの結果を図 3.2 および図 3.3 にそれぞれ示す．これらの図において，縦方向が N_1 の設定値，横方向が N_2 の設定値であり，該当した Web ページの数を各セルに記述した．

図 3.2，図 3.3 より，2 つの結果はほぼ同じ傾向であった．これらの図における N_1 と N_2 のピアソンの積率相関係数を算出すると，それぞれ 0.943 と 0.938 となった．このことから N_1 と N_2 は高い相関を持ち，ほぼ比例することが分かる．

(2) タグの深さの標準偏差と分割の閾値の関連

図 3.2 における $(N_1, N_2) = (6, 4), (4, 3)$ の場合について Web ページの特徴を比較した．結果を表 3.1 に示す．表 3.1 より，ファイルサイズがほぼ同等であるにも関わらず，その他の数値はいずれも $N_1 = 6$ の方が大きな値となった．このことから (N_1, N_2) と平均，標準偏差，コンテンツ数の統計値に相関があると考えられる．

以上の予備実験結果より，Web サイトごとに適切な N_1 と N_2 を表 3.1 で挙げた統計値を利用して動的に決定可能であると考えられる．ここで，N_1 と N_2 はコンテンツ間距離の差の大きさを判別する閾値であり，値のばらつきの度合いが影響することから，本研究ではこれらの統計値のうち意味が最も近い「標準偏差」に着目する．次に，閾値の動的決定方法について具体的に述べる．

(B) コンテンツ間距離の分布に基づく閾値 N_1, N_2 の動的決定手法

コンテンツ間距離の標準偏差に基づき，各 Web サイトごとに適切な N_1 と N_2 を動的に決定する手法を提案し，Web ページ動的分割アルゴリズムを示す．

任意の Web ページ t に適した閾値 N_{t1}, N_{t2} の算出手順を以下に示す．

(ア) 基準値を設定する．

(1) コンテンツ間距離の分布の差異に基づき閾値 N_{t1}, N_{t2} を決定するため，基準となる Web ページ (以下，基準ページと呼ぶ) を任意に選択する．

(2) 基準ページを最適に分割可能な N_1, N_2 を実験的に決定し，これを N_{b1}, N_{b2} とする．

(3) 基準ページのコンテンツ間距離 $S_{b(i,i+1)}$ の集合を 3.3.1 項の手順 (C) に従い導出する．

(4) 式 (3.2) に従い，σ_{S_b} を算出する．

		N_2																
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
N_1	0																	
	1																	
	2																	
	3			6														
	4			2	16													
	5				1	3												
	6				2	54	1											
	7						1	3										
	8					1		5										
	9								2									
	10									2								
	11								1		1	3						
	12																	
	13																	
	14																	
	15																	
	16											1						

図 3.2: N_1, N_2 の最適値の分布 (ニュース)

$$\sigma_{S_b} = \sqrt{\frac{\sum_{i=1}^{n_b-1} (\overline{S}_b - S_{b(i,i+1)})^2}{n_b - 1}} \tag{3.2}$$

$\overline{S}_b$：	基準ページのコンテンツ間距離の平均値
$S_{b(i,i+1)}$：	基準ページのコンテンツ i とコンテンツ $i+1$ 間の距離
n_b：	基準ページで抽出したコンテンツ数

(イ) 対象の Web ページに対して N_{t1}, N_{t2} を算出する.

		N_2																
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
N_1	0																	
	1																	
	2																	
	3			8														
	4			1	28													
	5				1	6												
	6					52	1											
	7							1										
	8						1	3	1									
	9																	
	10																	
	11																	
	12												1					
	13																	
	14																	
	15																	
	16																	

図 3.3: N_1, N_2 の最適値の分布 (金融)

表 3.1: $(N_1, N_2) = (6, 4), (4, 3)$ における Web ページの特徴比較

(N_1, N_2)	平均	標準偏差	コンテンツ数	ファイル サイズ
$(6, 4)$	9.5	23.7	1488.5	38.2
$(4, 3)$	4.7	10.9	1026.4	36.7

平均:タグの深さの平均値

標準偏差:タグの深さの標準偏差の平均値

コンテンツ数:コンテンツ数の平均値 (個)

ファイルサイズ:HTML ソースのファイルサイズの平均値 (Kbytes)

(1) 対象ページのコンテンツ間距離 $S_{t(i,i+1)}$ の集合を 3.3.1 項の手順 (C) に従い導出する.

(2) 式 (3.3) に従い，標準偏差 σ_{S_t} を算出する.

$$\sigma_{S_t} = \sqrt{\frac{\sum_{i=1}^{n-1} \left(\overline{S}_t - S_{t(i,i+1)}\right)^2}{n_t - 1}} \tag{3.3}$$

(3) 式 (3.4) および式 (3.5) に従い，N_{t1}, N_{t2} を算出する．ここで α は正の実数値 $(\alpha > 0)$ とし，$t = b$ のとき $N_{t1} = N_{b1}, N_{t2} = N_{b2}$ とする.

$$N_{t1} = N_{b1} + N_{b1} * \left(\frac{\sigma_{S_t}}{\sigma_{S_b}} - 1\right) * \alpha \tag{3.4}$$

$$N_{t2} = N_{b2} + N_{b2} * \left(\frac{\sigma_{S_t}}{\sigma_{S_b}} - 1\right) * \alpha \tag{3.5}$$

$\overline{S_t}$:	分割対象の Web ページの距離の平均値
$S_{t(i,i+1)}$:	分割対象の Web ページのコンテンツ i とコンテンツ $i+1$ 間の距離
n_t :	分割対象の Web ページで抽出したコンテンツ数

(C) 閾値の動的決定機能を有する分割アルゴリズム

提案した閾値の動的決定手法を組み込んだ Web ページ分割アルゴリズムを図 3.4 に示す．このアルゴリズムでは，3.3.1 項の手順に対し，N_{t1}, N_{t2} の導出手順を加えている.

[分割アルゴリズム]

$objectID = (root); x = LEFT;$
$D_{objectID,x} =$ (Whole of contents);
  $\sigma_{S_t} =$ Sdeviation{Standard deviation of content distances};
  $N_{t1} = N_{b1} + N_{b1} * (\sigma_{S_t}/\sigma_{S_b} - 1) * \alpha;$
  $N_{t2} = N_{b2} + N_{b2} * (\sigma_{S_t}/\sigma_{S_b} - 1) * \alpha;$
Segment($D_{objectID,x}$){
  $S_{\max} = \max\{S_{t(i,i+1)} : i \in D_{objectID,x}\};$
  $S_{\text{average}} = \text{average}\{S_{t(i,i+1)} : i \in D_{objectID,x}\};$
  $C_{\text{segmented}} =$ (Minimum number of tags in a group
      when it is segmented)
  if($S_{\max} > N_{t1} * S_{\text{average}}$){
    (Segment at the $S_{\max}$);
    $ChildObjectID =$ (objectID of created object by segmentation);
    Segment($D_{ChildObjectID,LEFT}$);
  };
  else if($S_{\max} > N_{t2} * S_{\text{average}}$ & $C_{\text{segmented}} > M$){
    (Segment at the $S_{\max}$);
    $ChildObjectID =$ (objectID of created object by segmentation);
    Segment($D_{ChildObjectID,LEFT}$);
  };
  if($x = LEFT$){
    Segment($D_{objectID,RIGHT}$);
    Return;
  };
  else if($x == RIGHT$ & $objectID \neq (root)$){
    Return;
  };
  else{
    END;
  };
};

図 3.4: N_{t1}, N_{t2} の動的算出機能を有する分割アルゴリズム

3.3.3 レイアウト情報とのハイブリッド化による分割精度向上手法

(A) レイアウト情報を利用した分割処理の必要性

3.3.1 項および 3.3.2 項により，タグ構造の複雑さに応じて適応的にパラメタを最適

化が可能な，タグの複雑さに基づくコンテンツ間距離を利用した Web ページの分割が行えるようになった．しかしながら，HTML ソースの記述順序と実際の Web ページのレイアウト位置の違いに起因する，以下の 2 つの問題が新たに生じる．

(問題 1) タグの複雑さに基づくコンテンツ間距離と，実際の Web ページ上での見た目のコンテンツ間距離が一致しない場合がある．具体的には図 3.5 に示すように，`<table>` タグでコンテンツのレイアウトを定義している場合に，1 行目の右端のセルに表示したコンテンツと，2 行目の左端のセルに表示したコンテンツを，上記の手法では隣接するコンテンツとして解釈してしまう．一方，Web ページ表示上の距離は明確に離れている．

(問題 2) ニュースサイトやポータルサイトの多くは，ヘッダーやフッター，サイドメニューなどの明確なコンテンツオブジェクトが含まれており，それぞれタグの構造の複雑さが異なる．しかしながら，上記の手法では Web ページ全体を対象としたパラメタ (N_1, N_2) 設定をしており，部分的な特徴に適応できていない．

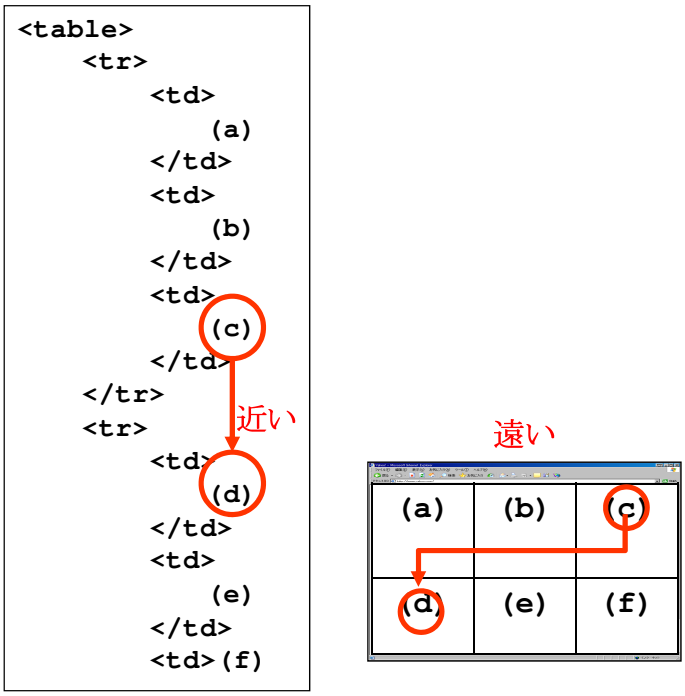


図 3.5: HTML ソース上の距離と Web ページ表示上の距離の違い

上記問題を解決するため，コンテンツ間距離とレイアウト情報の両方を利用したハイブリッド型の Web ページ分割手法に拡張し，分割精度向上を図る．この拡張手法は，次の 2 つのステップを前処理として実装することにより上記問題を解決する．まず，Web ページの大まかなレイアウトを把握するため，レイアウトに関連するタグを探し，初期分割を行うことで，問題 1 を解決する．Web ページのレイアウトに基づいているため，Web ページ上の実際に表示位置に合ったコンテンツオブジェクトに分割することが期待できる．次に，初期分割が行われた後のコンテンツオブジェクトに対してパラメタ N_1, N_2 を算出し，コンテンツ間距離に基づく分割処理を行うことで，問題 2 を解決する．コンテンツオブジェクトごとに最適なパラメタを設定するため，コンテンツオブジェクトごとのタグの複雑さが大きく異なる場合にも対応することができる．これにより，コンテンツ間距離に基づく分割処理において，高精度な分割結果を期待できる．しかしながら，これらの前処理を行うためには，HTML ソースのタグ構造が正しいことが必要であり，HTML ソースが正則であるという前提条件が必須となる．したがって，これらの前処理は HTML ソースのタグ構造が正しく解釈できた場合に限り，適用することとする．正しく解釈できない場合は，コンテンツ間距離に基づく分割処理のみを適用する．

(B) ハイブリッド型 Web 分割システムのアーキテクチャ

ハイブリッド型の Web ページ分割手法を実装したシステム (以下，ハイブリッド型 Web 分割システムと呼ぶ) の処理フローを図 3.6 に示す．本システムは次の 4 つのコンポーネントからなる．1 つ目は，Web ページのレイアウトを定義するタグを抽出するレイアウト分析機能 (Layout Analyzer)，2 つ目は，レイアウト分析機能の出力に対し，レイアウト情報に基づきコンテンツオブジェクトに分割する機能 (Layout based Segmentation Engine: LSE)，3 つ目は，コンテンツ間距離に基づきコンテンツオブジェクトを分割する機能 (Content-distance based Segmentation Engine: CSE)，4 つ目は，分割結果を携帯電話上での表示用に XHTML に再構築する機能 (XHTML Re-builder) である．

レイアウト分析機能において，対象となるタグとして，レイアウトのタグとしてよく利用されている `<table>`，`<div>`，`<frame>` に着目する．ここで，`<frame>` については，サブフレームとして異なる HTML ソースファイルが指定されており，明確に分離されていることから，個々の Web ページを分割する問題に帰着できる．よって，

本節では，<table> と をレイアウト情報の処理対象とする．

ハイブリッド型 Web 分割システムの処理フローの詳細を以下に示す．

(Step 1) Layout Analyzer が，対象の Web ページの HTML ソースファイルを取得し，<table> タグおよび タグを抽出する．

(Step 2) Layout Analyzer が，<table> タグと タグを正しい構造で抽出できた場合は，結果を LSE に入力する (Step 3 へ)．そうでない場合は，CSE に入力する (Step 4 へ)．

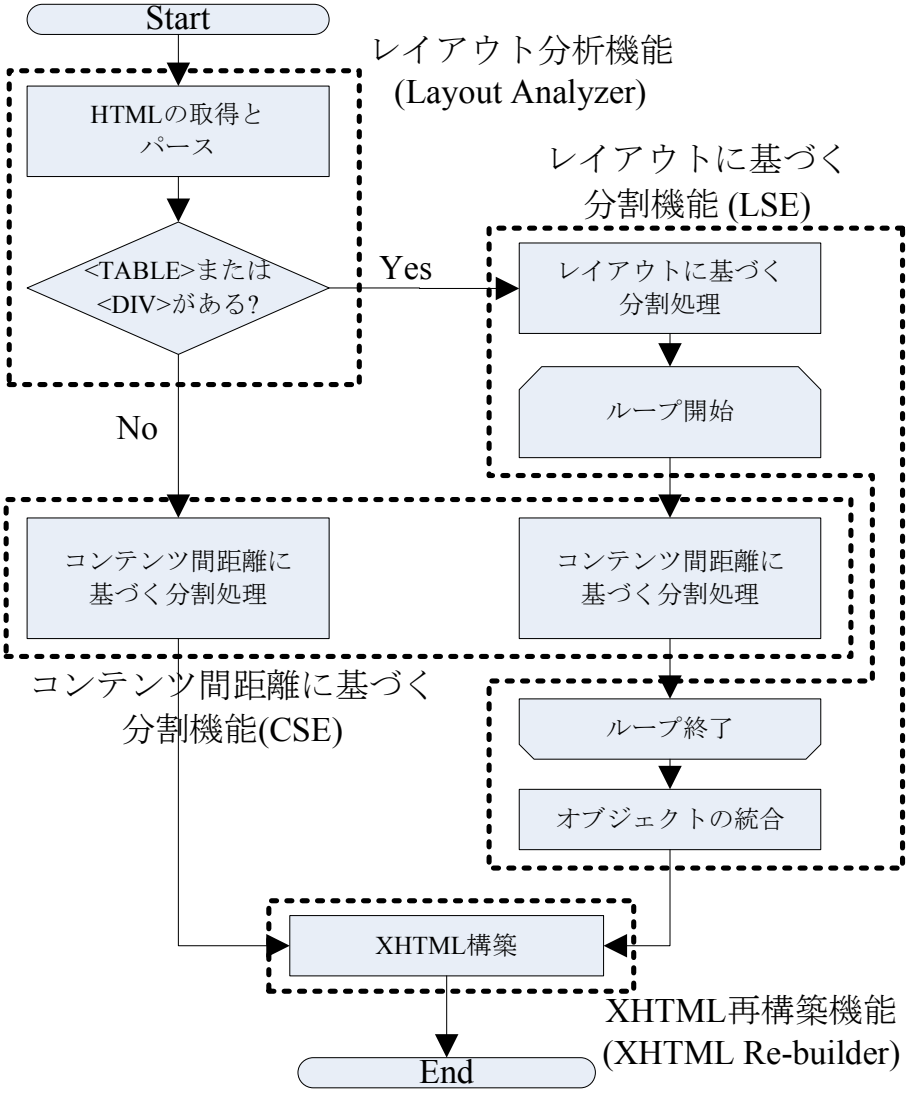


図 3.6: コンテンツ間距離とレイアウト情報のハイブリッド処理フロー

(Step 3) LSE は，HTML ソースを <table> タグと タグでコンテンツオブジェクトに分割し，サブオブジェクトごとに CSE に入力する (Step 4 へ)．LSE は CSE からの分割結果を返り値として受領し，全てのサブオブジェクトに対する CSE の分割結果を受領したら，結果をまとめて XHTML Re-builder に入力する (Step 5 へ)．LSE での分割方法の詳細は (C) で述べる．

(Step 4) CSE は，受領した HTML ソースまたはコンテンツオブジェクトをコンテンツ間距離に基づき分割する．Web ページの分割結果の場合は，XHTML Re-builder に入力する (Step 5 へ)．コンテンツオブジェクトの分割結果の場合は，LSE に応答する (Step 3 へ)．

(Step 5) XHTML Re-builder は，入力された分割結果をもとに，携帯向けの XHTML コンテンツを構築する．

(C) レイアウト情報に基づく Web ページ分割方法

<table> タグと，それに関連するタグ (<tr>,<td> など) は，表データを表示するためだけでなく，レイアウトを定義するためにも用いられている．よって，それらの 2 つの使い道を識別しなければならない．ここで，表データは比較的小さいセルが含まれることが想定されるのに対して，LSE の分割処理の目的は，大まかな Web ページのレイアウトを抽出することにある．そこで，分割可否の判定はセルの大きさに基づき判定する方針とし，その判定基準を比較的大きめに設定することで，おおまかなレイアウトを抽出しつつ，表データを分割しないようにする．また， タグについては，関連情報の集合を定義するだけでなく，スタイルファイルの適用範囲を指定するために利用する場合がある．いずれの場合においても，関連情報の集合として解釈することができるため，携帯電話の画面に合う大きさかどうかを判定すればよい．

そこで本節では，分割後のコンテンツオブジェクトの大きさを分割の主要な判定基準とすることで，Web ページの大まかなレイアウト情報を抽出する方針とする．ここで，コンテンツオブジェクトには，テキストだけでなく画像等も含まれることから，その大きさの定義が必要となる．本節では，大きさの単位を pixel に統一することとする．画像は pixel で定義されるため，画像サイズまたは HTML ソースで指定している表示サイズの値を利用する．テキストは，フォントの大きさと全角・半角の情報から pixel 数を計算する．コンテンツオブジェクトに分割するかどうかの基準値 ($P_{\max}$) を下回る場合は，分割は行わないとする．レイアウト情報に基づく分割の処理ステッ

プを以下に示す.

(Step 1) LSE が <table> タグと タグを HTML ソースファイルのポインタ位置から検索する．ポインタ位置の初期値は HTML ソースファイルの先頭とする．<table> タグまたは タグが見つかった場合は，その終了タグ (</table>,) にポインタ位置を設定し，Step 2 に進む．HTML ソースファイルの最後までタグが見つからない場合は，最後に検索した範囲をコンテンツオブジェクトとして CSE に入力し，処理を完了する．

(Step 2) LSE が <table> タグを検出すると，それに関連するタグ (<tr>, <td>, <thead>, <tfoot>, <th> など) を検出する．ここで，表データのヘッダー等を表すタグ (<thead>, <tfoot>, <th>) が検出された場合，LSE は表データであると判断し，分割は行わない (Step 1 へ)．検出した場合は，Step 3 へ進む．LSE が タグを検出した場合は Step 4 へ進む.

(Step 3) LSE が <tr> および <td> を検出し，それらのタグで区切られるセルの大きさの最小値を算出する．その最小値が $P_{\max}$ を下回る場合，LSE は分割を終了し，Step 1 に戻る.

(Step 4) タグで分割した結果の大きさが $P_{\max}$ 以下の場合，LSE は分割を終了し，Step 1 に戻る.

セルのサイズの算出方法は，セルの内部に含まれる画像やテキストの大きさを抽出し，以下のルールに従い算出する.

(1) LSE が <img>, <object>, <applet>, <layer>, <embed> などに含まれるサイズを表す属性値である “width” と “height” を抽出し，pixel 単位でサイズを算出する.

(2) 大きさを表す属性を持つタグ (<inputT>, <select> など) については，“size” や “maxlength” などの属性値やテキストの文字数から，pixel 単位のサイズを算出する．また，<input> で表示するボタン類については，“value” 属性値の文字数から算出する.

(3) テキストのサイズは，文字数とフォントサイズから算出する．フォントはすべて同一であるとし，フォントの違いによるサイズの差は小さいものとして無視する．また，空白文字についても無視する．LSE は，大まかなレイアウトを抽出することが目的であることから，$P_{\max}$ の値は Web ページ全体のサイズの 1/5

から 1/10 程度の大きい値を想定しており，これらの誤差が手法に与える影響は小さいと考えられる．

3.4 性能評価

提案手法の有効性を確認するため，評価実験を行った．本節では，その実験内容および結果について述べる．

3.4.1 コンテンツ間距離に基づく分割手法の性能評価

実験条件

(1) 評価対象の Web サイト

Chen ら [12] が評価対象とした Web サイトのうち，現在アクセス可能な 37 の Web サイトの各トップページを評価対象とする．Web サイト一覧を表 3.2 に示す．

(2) 比較対象

手法 A 論文に基づき独自に作成したパーサを利用して再現した Chen らの手法 [12]

手法 B-1 表 3.2 のリスト外の特定のページ [19] に対して適合率が最大化するように最適化した閾値を固定的に設定した Web ページ分割手法 [27]

手法 B-2 表 3.2 にリストした Web ページ対して適合率が最大になるように最適化した閾値を固定的に設定した Web ページ分割手法 [27]

手法 B-3 表 3.2 にリストした個々の Web ページの F 値が最大になるように最適化した閾値を個々に設定した Web ページ分割手法 [27]

提案手法 表 3.2 のリスト外の特定のページ [19] を基準ページとして，閾値を動的に決定する Web ページ分割手法

(3) 評価方法

各手法を PC 上に実装し，表 3.2 に示した Web サイトから HTML ソースを取得して各手法で分割処理を行った．各手法の分割結果について適合率と再現率を式 (3.6) および式 (3.7) に従い算出し，さらに F 値を算出して対象サイトの平均値を比較した．ここで「正解」とは，各手法で自動的に判定した個々の分割位置が，PC のブラウザ

表 3.2: 実験対象とする Web サイト

JobsOnline.com	Yahoo.com
flowgo.com	Msn.com
Americangreetings.com	Aol.com
Mypoints.net	Microsoft.com
Cnn.com	Altavista Search Services
Bizrate.com	Go.com
Mapquest.com	Amazon.com
Weather.com	Nbci.com
Infospace.com	Ebay.com
Iwin.com	Bluemountain.com
Espn.com	Lycos.com
Colonize.com	Looksmart.com
Travelocity.com	Cnet.com
Windowsmedia.com	Angelfire.com
Ivillage.com	Tripod.com
Disney Online	Iwon.com
Zmedia.com	Zdnet.com
Google.com	Msnbc.com
Earthlink.net	

で対象の Web サイトを見た評価者が主観により最適と判定した分割位置の何れかと合致することを指す．手法 B-1，手法 B-2，手法 B-3，および提案手法の各パラメタの設定値を表 3.3 に示す．

$$\text{適合率} = \frac{(a)\text{ 正解した分割位置数}}{(b)\text{ 全分割位置数}} \tag{3.6}$$

$$\text{再現率} = \frac{(a)\text{ 正解した分割位置数}}{(c)\text{ 全正解分割位置数}} \tag{3.7}$$

(a) 正解した分割位置数:	各手法の分割位置のうち正解した数
(b) 全分割位置数:	各手法で分割した位置数
(c) 全正解分割位置数:	人手で判断した正解位置数

表 3.3: パラメタ設定値

手法	パラメタと値
手法 B-1	$N_1 = 2.6, N_2 = 1.7, M = 2$
手法 B-2	$N_1 = 2.9, N_2 = 2.6, M = 2$
手法 B-3	N_1, N_2 は各 Web ページの最適値, $M = 2$
提案手法	$N_{b_1} = 3.4, N_{b_2} = 2.3, M = 2, \alpha = 0.36$

実験結果

(ア) 分割性能比較

実験結果を表 3.4 に示す．表 3.4 より手法 B-1 では F 値が 0.59 となり，手法 A の 0.45 よりも高い結果となった．但し，手法 A の結果については著者らが Chen らの手法の論文 [12] から読み取れる情報に基づき実装しているため，Chen らの実装とは異なる可能性もあることから参考値とする．閾値を最適化した手法 B-2 では F 値が 0.58 となり手法 B とほとんど違いは得られなかった．実験では適合率を最大化するようにパラメタ調整しているため適合率で比較すると，手法 B-1 が 0.64 であるのに対し，手法 B-2 は 0.80 と大幅に上回っていることがわかる．また，提案手法では F 値が 0.64 となり手法 B-2 よりもさらに高い値が得られた．また，手法 B-3 の結果は提案手法の理想値に等しい意味を持つが，提案手法の結果はその理想値に近い値が得られた．以上の結果から，コンテンツ間距離の標準偏差に基づく閾値の動的決定が有効に作用していることを示し，提案手法の有効性を確認できた．

(イ) 考察

提案手法では，対象とした表 3.2 の Web ページの中で，aol.com の Web ページの再現率が最も低い結果となった．18 の正解位置数に対して 1 箇所しか分割位置を検出できていないことが原因であり，再現率は 0.06 となっていた．この理由として，提案手法の特徴はレイアウト用などの直接表示されないタグに着目して分割処理を行う点であることから，そ

表 3.4: 実験結果

手法	適合率	再現率	F 値
手法 A	0.71	0.33	0.45
手法 B-1	0.64	0.55	0.59
手法 B-2	0.80	0.45	0.58
手法 B-3	0.72	0.61	0.66
提案手法	0.87	0.50	0.64

れらのタグがない，あるいは少ない場合には分割が難しい結果となると考えられる．実際に，aol.com のページの HTML ソースを分析すると，コンテンツのレイアウトをほぼ完全に CSS (Cascading Style Sheets) ファイルで記述しており，HTML ソース自体は非常にシンプルに記述されていた．このとき，提案手法はコンテンツ間距離をほとんど 0 と算出してしまうため，分割する点をうまく抽出することができなかった．

3.4.2 ハイブリッド手法の評価

コンテンツ間距離に加えてレイアウト情報を利用したハイブリッド手法の性能を検証するため，性能評価を行った．実験条件と結果を以下に述べる．

実験条件

レイアウト情報を加えることによる効果を検証するために，コンテンツ間距離のみを利用した結果と比較する．以降，コンテンツ間距離による分割手法を CD (Content Distance) 手法と呼び，ハイブリッド手法を HYB (HYBrid) とそれぞれ呼ぶこととする．評価対象の Web サイトとして，英語サイトと日本語サイトから合計 100 サイトを選択した．選択方法は，Web サイトのアクセス数ランキング情報を提示している Alexa[4] において，ランキング上位に位置する Web サイトを参考にし，Javascript や CSS を多用したサイトは除いた．サイトの一部を表 3.5 に示す．また，パラメタの設定値を表 3.6 に示す．CD 手法と HYB 手法をそれぞれ実装し，評価対象の Web ページから取得した HTML ソースファイルを両手法に入力した．

精度評価の指標として，適合率と再現率を測定し，F 値を算出した．適合率と再現率はそ

表 3.5: 評価対象の Web サイトの例 (番号は Alexa が提示している順位とは無関係である)

	英語サイト		日本語サイト
1	Yahoo!	1	Yahoo! Japan
2	Google	2	Google Japan
3	Myspace	3	Mixi
4	MSN	4	Rakuten Ichiba
5	Ebay	5	Livedoor
6	Amazon	6	FC2
7	Youtube	7	Goo Dictionary
8	Craigslist	8	Amazon JP
9	Wikipedia	9	Infoseek Japan
10	CNN	10	@nifty
11	LinkShare	11	MSN Japan
12	Thefacebook	12	Wikipedia
13	TypePad	13	Asahi.com
14	Blogger	14	DION
15	Target	15	Hatena

表 3.6: パラメタ設定値

N_{b1}	N_{b2}	M	α	$P_{\max}$
3.4	2.3	2	0.36	156000

れぞれ式 (3.6) および式 (3.7) を用いて算出した．なお，(*c*) 全正解分割位置数については，評価対象の Web ページに対し，人手で正解の分割位置を付与した結果を用いている．

結果

2 手法の比較結果を表 3.7 に示す．また，HYB 手法の F 値が高い場合，および結果が変わらなかった典型的な結果の例を表 3.8 に示す．

表 3.7 より，英語サイトにおいて F 値が 0.63 から 0.73 に改善し，また日本語サイトにおいて F 値が 0.71 から 0.75 に改善し，HYB 手法が CD 手法と比較して高い F 値を達成した．これにより，レイアウト情報を利用することで，提案手法がより高精度になることを示した．なお，英語サイトと日本語サイトについて，結果において特に大きな差は見られなかった．また，表 3.8 に典型的な結果の例を示し，実際の分割イメージを図 3.7 および図 3.8 に示す．図において，破線が CD 手法を適用した場合の分割結果であり，実線が HYB 手法の分割結果を表す．前者の例は，HYB 手法が有効に作用した例であり，Target.com 社のサイト [59] に適用すると，F 値が 0.35 から 0.96 に大幅に改善した．図ほとんどの正解分割位置で分割できた．後者の例は，HYB 手法がうまく作用しなかった例であり，LinkShare 社のサイト [47] に適用した場合に，正解分割位置をより多く抽出できた個所がある一方で，CD 手法で抽出していた正解分割位置を抽出できなかった箇所があり，F 値が変わらない結果となった．

後者の例においてうまくいかなかった理由としては，$P_{\max}$ の設定値が適切でなかったことが考えられる．LinkShare 社のサイトは，主に `<table>` タグでレイアウトが構築されており，シンプルな HTML ソースとなっている．そのため，タグ構造の複雑さを利用するコンテンツ間距離での分割が，レイアウトに基づく分割後にうまく働かなかったと考えられる．

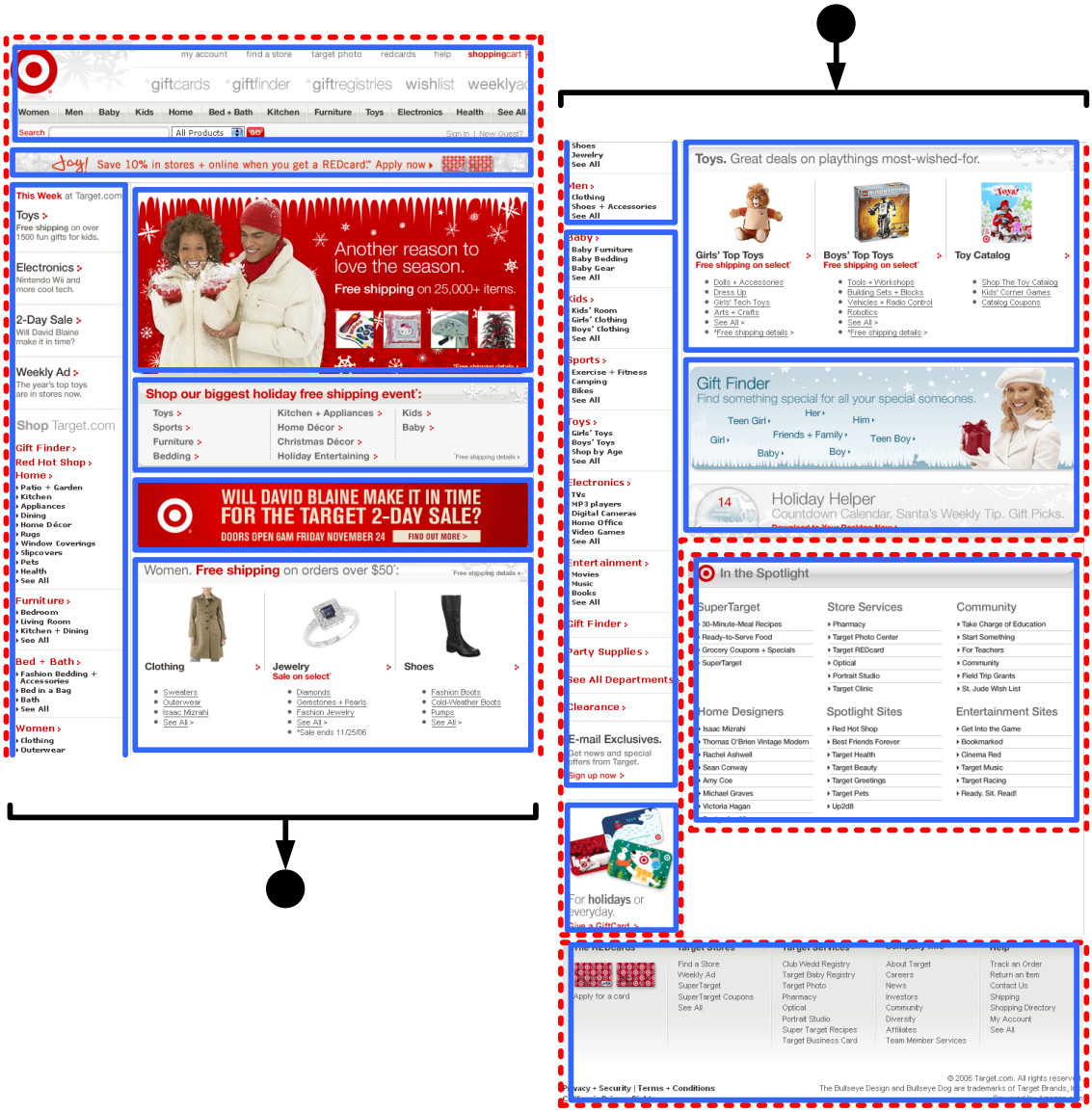
表 3.7: 分割精度の評価結果 (平均値)

	手法	適合率	再現率	F 値
英語サイト	CD 手法	0.74	0.55	0.63
	HYB 手法	0.77	0.70	0.73
日本語サイト	CD 手法	0.82	0.62	0.71
	HYB 手法	0.82	0.69	0.75

表 3.8: 典型的な結果の例

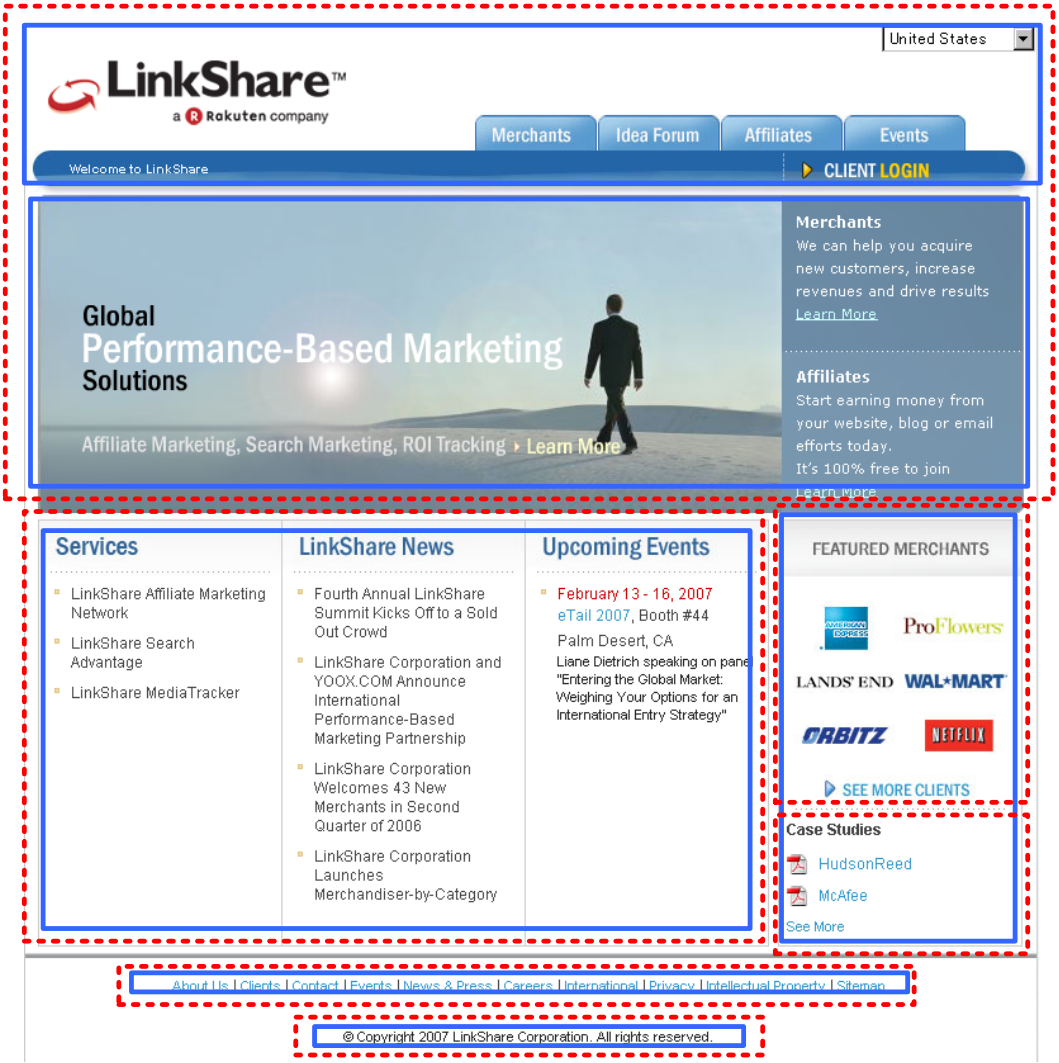
	Web サイト	F 値 (CD 手法)	F 値 (HYB 手法)
有効に作用した	Target(図 3.7)	0.35	0.96
有効に作用しなかった	LinkShare(図 3.8)	0.77	0.77

この点については，レイアウトに基づく分割において，タグ構造の複雑さに応じて，$P_{\max}$ の設定値を適応的に変えるなどにより，改善できると考えられる．



CD方式:破線, HYB方式:実線

図 3.7: HYB 手法で良好な分割結果が得られた例 (`http://www.target.com/`)



CD方式:破線, HYB方式:実線

図 3.8: HYB 手法で誤りを含む分割結果の例 (`http://www.linkshare.com/`)

3.5 実装例とユーザビリティ評価

提案手法のユーザビリティ評価のため，HYB 手法を携帯電話上で動作するように実装した．Google Wireless Transcoder[23] を比較対象として，操作性に関する比較評価を行った．さらに，提案手法のサーバ上での処理時間についても評価を行った．

3.5.1 実装例

HYB 手法の実装例として，携帯電話向け Web 閲覧システムを図 3.9 に示す．本システムは，(1) ユーザからの要求を受け付ける要求受付機能，(2) 要求に応じて対象 Web ページの HTML ソースを取得するブラウザエミュレータ，(3) 取得した HTML ソースを提案手法により分割する分割処理機能，(4) 分割処理結果を再構成して新たな XHTML ソースを生成する XHTML 構築機能の 4 つから構成される．

図 3.10 に示した写真つきのニュースを提供している Web ページ [39] に対して本システムを適用すると，例えば破線で囲まれた写真とテキストの組を 1 つのコンテンツオブジェクトとして正しく認識し，図 3.11 に示すように，コンテンツオブジェクトのリストを生成できた．さらに，コンテンツオブジェクトのリストはそれぞれハイパーリンクになっており，その 1 つを選択することで情報にたどり着くことができた．このシステムを利用することで，ユーザは，少ないスクロール操作とクリック操作により，必要な情報を閲覧することができるようになる．

3.5.2 ユーザビリティ評価

実験方法

システムのユーザビリティの評価を行うため，シミュレーション評価実験を行い，HYB 手法と Google Wireless Transcoder(以下，GWT 手法と呼ぶ) を比較した．ここでは，ユー

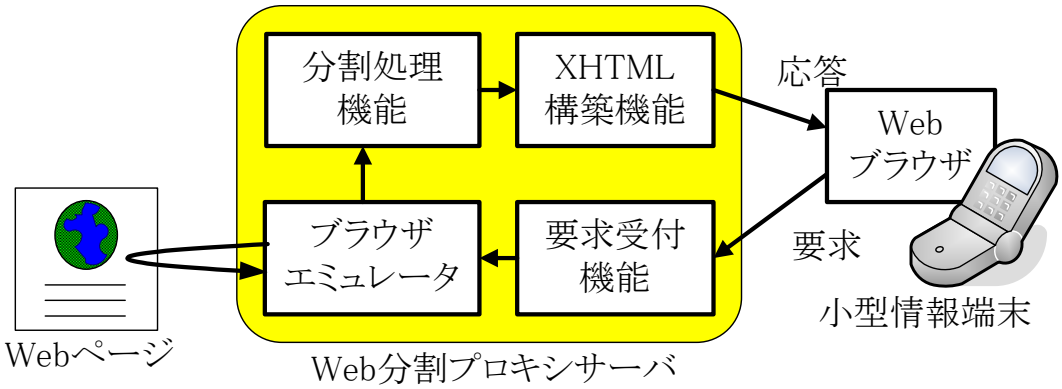


図 3.9: 携帯電話向け Web 情報閲覧システム

図 3.10: HYB 手法を適用した分割結果の例 (http://www.japantimes.co.jp/)

ザが携帯電話を利用して，Web ページ中のある表示位置にある情報を探して閲覧する状況を想定した．評価方法は，初期画面が表示されてから，ユーザが指定された情報にたどり着くまでの推定時間を測定する．ここで，推定時間とは，スクロールのためのボタン操作回数と，リンクをたどるためのボタン操作回数に対し，単位回数当たりの時間を掛けることで時間に換算して算出する値とする．なお単位回数当たりの時間は，スクロール操作については方向キーをクリックするような操作を想定し，また通信を伴わない操作であるため，1 回あたり 0.2 秒とした．また，リンクをたどる操作については，ボタンを押してから通信を行い，画面上に表示するまでの時間として，1 回あたり 5.0 秒とした．これらの値は，

コンテンツオブジェクトのリスト　　コンテンツオブジェクト

図 3.11: 携帯電話上での表示例

実際の携帯電話を利用して，主観的に定義した値である．

対象とした Web サイトは，表 3.5 の日本語サイトのうち，情報量の多い 1, 4, 5, 13, 14 の 5 つのサイトを対象とした．さらに，閲覧のターゲットとなるコンテンツを，各 Web ページの先頭付近 (Top)，中央付近 (Middle)，末尾付近 (Bottom) にあるコンテンツに設定した．それぞれについて操作の推定時間を測定し，5 つのサイトの平均値を算出して比較する．

結果

評価結果を図 3.12 に示す．図において，横軸はターゲットのコンテンツの位置 (Top, Middle, Bottom) を表し，縦軸はターゲットのコンテンツが表示されるまでの推定時間を表す．この結果より，Bottom の場合において，HYB 手法が GWT 手法と比較して，約 1/4 の時間でターゲットのコンテンツにたどり着けることがわかった．また，Middle の場合は約 1/2 であり，Top の場合はほぼ同等となった．以上により，提案した HYB 手法は，GWT 手法と比較して，ターゲットのコンテンツが表示されるまでの推定時間において，大幅に改善

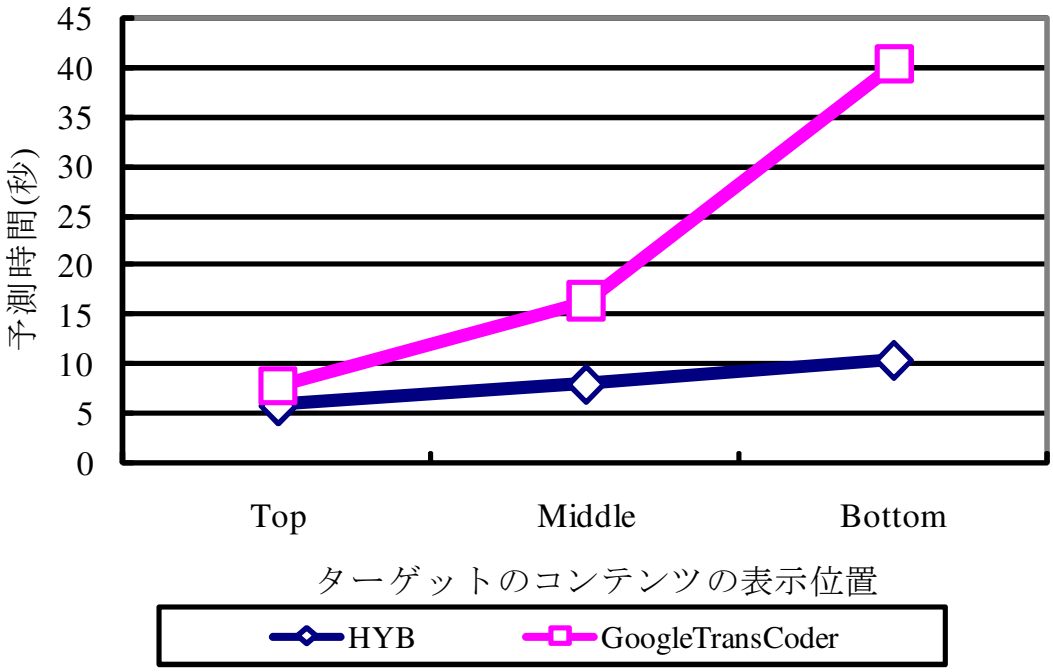


図 3.12: ユーザビリティ評価結果

できたといえる.

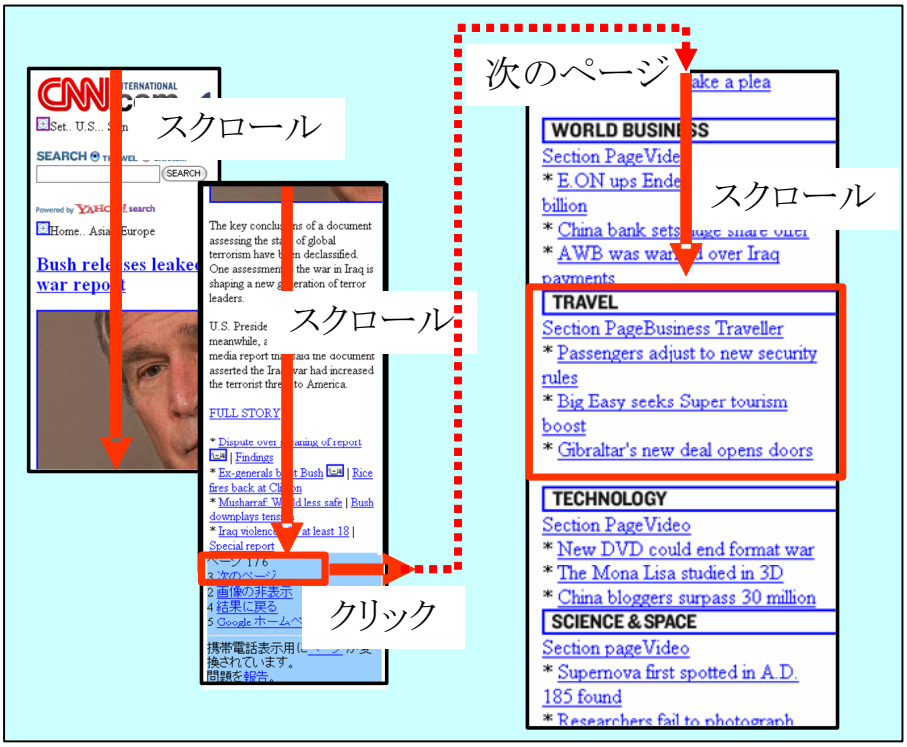
具体的な比較の例を図 3.13 に示す. ここでは, CNN のサイト [16] に対し, HYB 手法と GWT 手法を適用した場合のユーザの操作イメージを示している. ユーザが “TRAVEL” のカテゴリの情報を閲覧しようとしている状況において, 明らかに, HYB 手法の方が操作数が大幅に少なくなることが分かる.

3.5.3 処理時間の検証

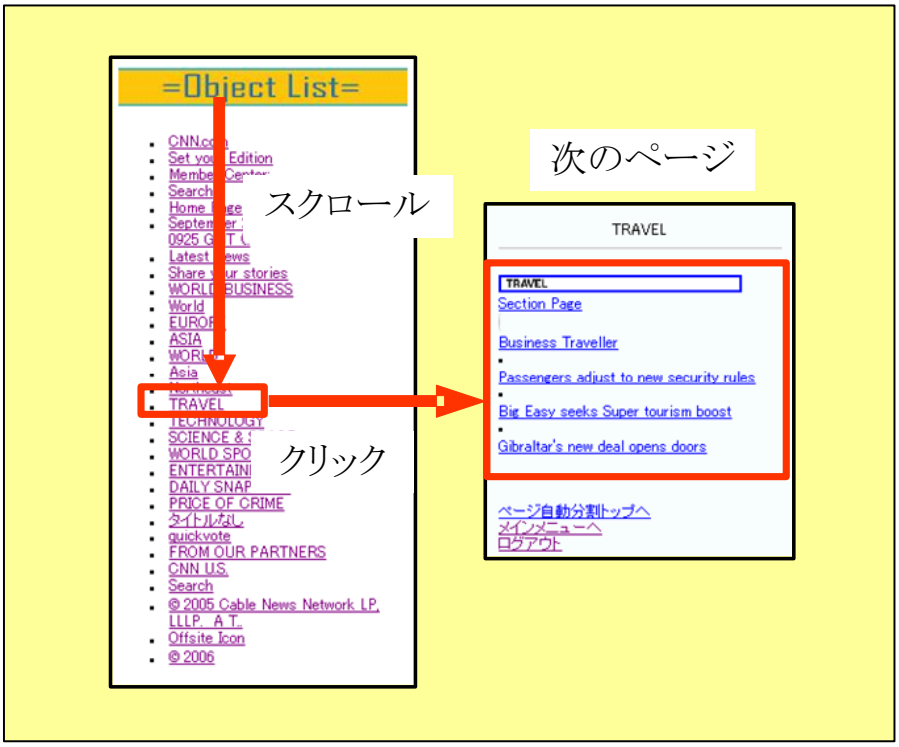
HYB 手法のサーバ上での処理時間を測定した. 以下に実験環境と評価結果を述べる.

実験環境

サーバを 1 台 (CPU クロック数:3.4GHz, メモリ:2GB) 用意し, HYB 手法を実装した. 対象の Web ページを表 3.5 から任意に 4 サイトを選択した. 以下の主要なプロセス (a) から (f) について, それらのサーバ上での処理時間を測定した. なお, 通信時間は場所や時刻により大きく変化するため, 測定値には含めていない.



Google Wireless Transcoder



提案方式

図 3.13: ユーザビリティ評価結果の具体例 (`http://www.cnn.com/`)

(a) 携帯電話からの URL 要求受付処理
(b) HTML ソースファイルのパース処理
(c) 画像のスケール処理
(d) タグの深さを抽出する処理
(e) HYB 分割処理
(f) XHTML の再構築処理

結果

測定結果を表 3.9 に示す．この結果より，処理時間が長いのは (b) と (c) であることが分かる．分割処理は，例えば日本語サイト 4 においては合計の 10%程度 (180.7/1838.6 × 100) であり，さほど重い処理ではなことが分かる．他の 3 つの結果についてもほぼ同様である．また，合計のばらつきは，(b) と (c) の処理時間の影響が大きい．つまりばらつきの要因は，HTML ソースファイルの長さと Web ページに含まれる画像の数であった．

3.6 むすび

本章では，PC 向けの Web ページを小分割して Web ページを再構成し，画面の小さい携帯電話でも容易に閲覧するためのコンテンツ間距離に基づく Web ページ自動分割手法について検討した．ここで，Web ページ毎に適切な分割閾値をコンテンツ間距離の標準偏差に基づき動的に決定する手法，さらに，レイアウト情報を利用した分割処理を加えた，ハイブリッド型の分割手法を提案した．提案手法の有効性を示すため，分割精度について従来

表 3.9: 処理時間の評価結果 (msec)

	(a)	(b)	(c)	(d)	(e)	(f)	合計
日本語サイト 1	49.5	921.8	379.4	11.1	211.2	3.1	1576.1
日本語サイト 4	48.7	857.5	743.8	6.3	180.7	1.8	1838.6
日本語サイト 6	47.0	697.9	826.0	72.6	135.1	1.8	1780.3
日本語サイト 14	44.8	280.6	216.5	11.2	61.1	1.8	616.0

手法と比較する評価実験を行った．コンテンツ間距離に基づく分割処理の評価実験では，F 値が 0.64 で従来手法よりも高い値が得られ，手法の有効性を示した．また，ハイブリッド型の分割処理の評価実験では，さらなる分割精度の向上を実現できており，その有効性を確認した．

本研究成果を応用した Web ページ自動分割システムは，KDDI 株式会社 [44] が運営するベータ版サービス提供サイトである「au one ラボ [6]」において，au one ポータル [5] の各サービスの Web ページを対象とした「携帯向けホームページダイジェスト表示ツール」として一般公開している．

今後の課題として，分割精度向上のための改善が挙げられる．タグの意味やスタイル情報，画像の意味などの，より深い意味を抽出して利用するなど，HTML ソースに含まれる意味情報を活用した新たな方式を検討する必要がある．

第4章

小型情報端末向けクリック型Web検索手法

4.1 まえがき

インターネット接続可能な小型情報端末が普及しており，PC 以外の Web 閲覧環境が増加している．ユーザは閲覧する場所を限定されることなく，また PC の操作に詳しいユーザでなくても，誰でも気軽にインターネットの情報にアクセスできる環境が整いつつある．

しかしながら，これらの端末で Web 閲覧を行うためには，ユーザは数字キーや方向キーなどの限られた入力インタフェース (以降，リモコン型 UI と呼ぶ) を利用することになる．このため，キーボードやマウスを有する PC と同様の操作性を実現するのは難しい．ここで，マウスについては，一部の携帯電話端末にマウスポインタが搭載されていることや，携帯電話上でマウスを動作させるソフトウェア [64] が開発されていること，さらに一部の TV ゲーム機のブラウザではポインタが利用できるなど，ポインタの搭載は進んでいる．一方，キーボードについては，個々のキーを指で押す必要があることから，ある程度の設置面積が必要であるといった物理的な制約がある．そのため，一部の高機能な携帯電話端末や PC 一体型のテレビ端末に搭載されているのみであり，ほとんどの端末はキーボードを搭載していない．画面上に表示されたソフトウェアキーボードをポインタでクリックしながら文字入力する手段はあるが，物理的なキーボードほどの快適な文字入力はできない．

一方，多くの Web 検索サービスは日々進化しており，少ない検索語数でもまったく見当違いの Web ページが提示されることは少なくなった．しかしながら，基本的には多くの人が望むと推定した Web ページを優先的に提示するアルゴリズムであるため，ユーザが稀な情報を探している場合は，複数の適切な検索語を選び，試行錯誤をしながら目的の情報を

探し当てる作業を行うことになる．さらに，OneStat.com 社の調査結果によると [57]，Web 検索時に入力する検索語数で利用頻度が最も多いのは 2 語であり，また 10 語までの頻度分布で平均すると約 3 語となる．よって，望みの情報を得るためには多くの文字を入力する必要があるといえる．

以上のことから，リモコン型 UI のみの小型情報端末で，一般的な Web 検索操作を満足に行うことは非常に困難であるといえる．よって，リモコン型 UI を利用した Web 検索を容易に行えるようにするために，検索語入力の手間を削減する新たな Web 検索手法を検討する必要がある．ここで，ユーザの検索行動を，任意の語で検索する場合と，表示中の情報に関連する語で検索する場合を組み合わせた行動であると仮定する．このとき，前者についてはユーザの頭に浮かんだ検索語を入力する必要があるため，文字入力操作は避けられない．一方，後者については，検索の足掛かりとなる語が必ず Web ページ中に表示されていることとなり，ユーザの検索語入力操作を削減することができる可能性が高い．

そこで本章では，ユーザが Web ページ上で注目した語 (以降，注目語と呼ぶ) をポインタでクリック操作をすると，その前後にある語 (以降，周辺語と呼ぶ) の重要度に基づき検索語を動的に生成して検索するクリック型検索手法を提案する．本手法は，周辺語の重要度を，品詞や意味，表示方法等に基づく語の単独での重要度と，注目語に対する関連の強さに基づく重要度の 2 つの指標を用いて計算する．

以下，4.2 節では，キーボードを多用しない Web 検索手法に関連する既存研究について述べる．4.3 節では，小型情報端末のためのクリック型 Web 検索システムの機能要件を説明する．4.4 節では，提案方式である周辺語の重要度モデルに基づくクリック型検索手法を提案する．4.5 節では，提案方式の性能評価を行い，最後に，4.6 節で本章のまとめを述べる．

4.2 関連研究

重要語の推定手法ならびにキーボードを多用しない検索手法について，関連研究とその問題点を述べる．

重要語の推定手法として，語の共起情報を利用して重要な語を文書から抽出する手法 [50] や，多義語に代表される検索語の曖昧性を解消するために，検索結果から重要語を抽出する手法 [62] がある．前者は，コーパスや単純なルールを利用する重要語推定方式 [20, 40, 46] では Web 文書の多様性には対応できないことを考慮として，Web 文書単独での重要語推定方式を提案している．また後者は，一部の語とだけ共起する語を重要語とする手法であ

る．これらの手法は単独あるいは限られた Web ページ集合から重要度を推定する点で，本研究が着目している課題と類似しており，語の重要度の 1 つの要素として活用することができる．しかしながら，これらが抽出しているのは 1 つの Web 文書あるいは限られた Web 文書集合内における重要度のみであり，ユーザが着目した語に対する重要度に関する検討がないという問題がある．

ペンインタフェースと検索目的指定による検索操作補助手法 [36] は，タッチパネルとペンによる入力インタフェースを利用して，ユーザがペンのドラッグ操作で検索語を選択すると，システムがその周辺にある語の種別 (企業，施設など) に応じた検索意図メニュー (地図が見たい，ニュースリリースが見たいなど) をユーザに提示する．この方式は，ユーザの検索意図を確実に絞り込むことができる利点はあるが，リモコン型 UI では正確に文字列をなぞるのが難しいことや，検索に至るまでの入力の手数が多いこと，さらに語の種別を判別するための辞書および検索意図知識をあらかじめ構築しておき，定期的に更新するなどの対応が必要であることが問題である．

また，検索結果を分類しながら絞り込んでいく検索手法 [18, 43] は，検索結果を内容に基づくいくつかのグループに分類し，ユーザがそのうちの 1 つを選択することで検索目的を伝える方式である．グループを選択するとさらにその中の Web ページを詳細なグループに分類していき，このインタラクション操作を繰り返すことで絞り込みを進めていく．しかしながら，検索目的を段階的に指定していくことになるため，検索結果が絞り込まれるまでにはインタラクションが何度も必要となり，検索結果の数によっては入力の手数が大きくなるという問題がある．

4.3 クリック型 Web 検索システムの概要と機能要件

本章では，ポインタが利用できることを前提とし，ユーザがリモコン型 UI を利用して，容易に検索操作ができるシステムの実現を目標とする．ここで，リモコン型 UI の操作性については，たとえば赤外線や加速度センサを利用したポインティングデバイスの場合は，画面上の任意の位置に素早くポインタを移動することが出来る利点がある．しかしながら，ドラッグ操作で必要なポインタの位置を正確に水平または垂直方向に移動させることや，一定位置にポインタを維持することが難しいという欠点がある．また，携帯電話に適用されている方向キーを利用したポインティングデバイスの場合は，ポインタの位置を正確に移動・固定することが出来る利点があるが，任意の位置に素早く移動させることが難しいと

いう欠点がある．

以上の検討結果から，リモコン型 UI 向けのクリック型 Web 検索システムは，検索に必要なクリック数を最小限にすることが機能要件となる．本研究では，ユーザが指定した語に対し，その語を修飾できる「重要語」を周辺から判別して，適切な検索語を自動生成することで機能要件を満足する方針とする．まず，最小のクリック数である 1 回のクリックで検索する方式について提案し，基本的な性能評価を行う．次に，2 クリック以上を用いて検索の満足度を上げる方法について，実装方法の検討と合わせて議論する．

想定するクリック型 Web 検索システムの利用シーンを図 4.1 に示し，ユーザの検索操作手順を以下の (1)〜(3) に示す．

(1) ある Web ページが画面上に表示されているとする．この Web ページはユーザのブックマークやポータル設定，あるいはそれらからリンクを辿る等の手順を行い，表示されているものとする．

(2) 画面上に表示されているテキストの中に検索したい語があった場合に，ユーザはポインタをその語に重ねるように操作し，次にボタンをクリックする．これにより，ユーザはシステムに対し検索語を指定する．

(3) リスト形式やサムネイル表示，あるいは最適と推定した Web ページを直接表示す

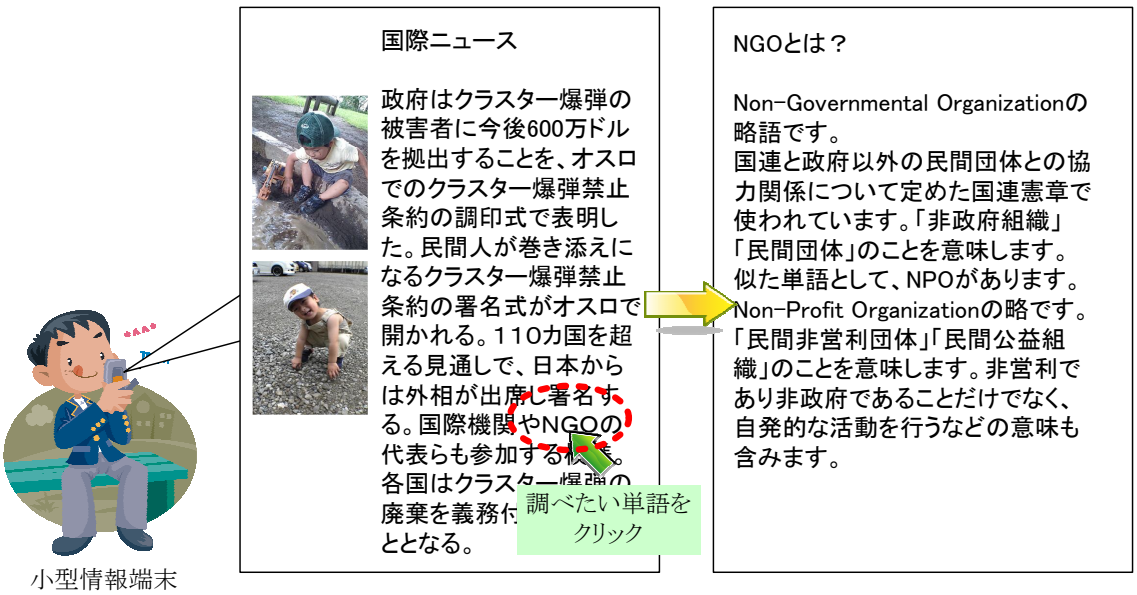


図 4.1: リモコン型 UI を利用した検索語指定操作の例

るなど，何らかの形式で検索結果が画面上に表示される．ユーザは提示された検索結果の中から望みの Web ページを探す．

4.4 周辺語重要度モデルに基づく Web 検索手法の提案

4.3 節で述べた機能要件を満たす，リモコン型 UI 向けのクリック型検索手法を提案する．クリック数を最小限にするため，1 回のクリックで最適と思われる 1 つの Web ページを提示する提示型ユーザインタフェース [53] とし，それ以外の検索結果を参照する場合は，クリック操作のみで順次 Web ページを提示する方針とする．また，クリック数を削減するため，検索結果を後で絞り込む方針 [18, 62] ではなく，検索語を自動的に追加して絞り込む方針 [53] とする．ここで，周辺語は注目語を修飾する重要な語であると仮定し，ユーザが指定した注目語の周辺に表示されている周辺語から重要度の高い語を選択して検索語を自動生成する．以下に，クリック型検索手法の処理フローと周辺語を選択する指標である重要度モデルについて述べる．

4.4.1 クリック型検索手法

提案方式の処理フローを図 4.2 に示す．ここで (Step3) で抽出する周辺語数は予備実験を行い決定する．図 4.2 の各ステップについて詳細を以下に述べる．

(Step1) クリック位置の前後にある文字列の取得
ユーザがクリックした位置に最も近い文字を中心として，前後 n 文字を切り出す．

(Step2) 自立語・複合語の抽出
形態素解析ツール [15] を用いて名詞に属する語を抽出する．また，複合語の生成ルールとして，連続する名詞は複合語とし，特定の接尾辞の前では区切ることとする．ここで特定の接尾辞とは，「サ変接続，一般，形容動詞語幹，助数詞，助動詞語幹，人名，地域，特殊，副詞可能」である．また，切り出した文字列の両端にある語は，不完全である可能性が高いため，ここで除外する．

(Step3) 注目語・周辺語の抽出
(Step2) で抽出した語の中から，クリック位置に表示されている語を注目語とし，それ以外の語を周辺語とする．

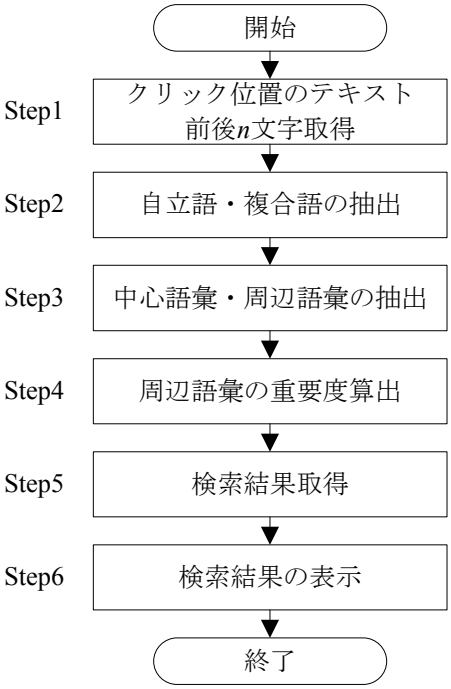


図 4.2: 処理フロー

(Step4) 周辺語の重要度算出
抽出した周辺語について，注目語と組み合わせて検索することで絞り込みができるような語を選択する．

(Step5) 検索結果取得
注目語と，(Step4) で選択した周辺語を組み合わせて，一般の検索サイトが提供している検索 API[67] を利用して検索結果を取得する．

(Step6) 検索結果の表示
検索結果の Web ページを上位から順にユーザの端末に表示する．

4.4.2 周辺語の重要度モデル

ある周辺語について，注目語に組み合わせて検索語とすることで，検索結果の絞り込みができる可能性の度合いを「周辺語重要度」と定義する．周辺語は注目語の意味を補完できる関連性の高い語である必要がある．そこで，

(1) 語自体の重要度：その語が単独で重要な意味を持つこと
(2) 注目語に対する重要度：注目語との関連性が高いこと
の 2 つの指標で周辺語を評価し，その加算値が高い値を持つ場合に周辺語重要度が高いと判定するモデルを提案する．提案モデルを式 (4.1) に示す．

式 (4.1) において，注目語 w_a に対する周辺語 w の周辺語重要度 $E(w, w_a)$ は，第 1 項の w における語自体の重要度 $E_o(w)$ と，第 2 項の w_a に対する w の重要度 $E_r(w, w_a)$ の重み和とする．また，α は第 1 項と第 2 項の加算比率を表す重み係数である．

$$E(w, w_a) = \underbrace{\alpha \cdot Eo(w)}_{\text{第 1 項}} + \underbrace{(1-\alpha) \cdot Er(w, w_a)}_{\text{第 2 項}} \tag{4.1}$$

4.4.3 重要度の算出

$E_o(w)$ および $E_r(w, w_a)$ を測定するための，具体的な要素の例を表 4.1 に示す．ここでは，ニュースや史実などの文書に含まれる注目語を対象とし，周辺語として関連する人物や事象の場所などの固有名詞が利用可能である状況を想定する．$E_o(w)$ の要素として，地名や人名などの文書の内容を限定しやすい固有名詞であるか否かのフラグ $F_m(w)$，および強調表現などの Web ページの著者が意図的に重要であると表現した語であるか否かのフラグ $F_v(w)$ の 2 つとする．また，$E_r(w, w_a)$ の要素として，注目語との共起の度合い $F_c(w, w_a)$，および注目語との間の文字数に基づき算出する距離 $F_d(w, w_a)$ の 2 つとする．

以下に，表 4.1 の要素を持つ場合の式 (4.1) の第 1 項および第 2 項の算出例を述べる．
(1)$E_o(w)$ の算出

$E_o(w)$ は，$F_m(w)$，$F_v(w)$ を含む式 (4.2) に基づき算出する．$p(w)$ は w の統計的な重要度を表す関数とし，ここでは，語の重要度の指標の 1 つである IDF (Inverse Document Frequency)[58] を利用する．IDF はその語が出現する Web ページ数が少ないほど高い値となり，その語の専門性を示す指標となる．これにより，あらゆる Web ページに出現するような，情報の絞り込みに寄与しない語の重要度を下げることができる．

$$E_o(w) = (F_m(w) + F_v(w) + 1) \cdot p(w) \tag{4.2}$$

表 4.1: 周辺語重要度の要素一覧

重要度種別	要素
$E_o(w)$:語自体の重要度	$F_m(w)$:意味 (地名，人名など) $F_v(w)$:表示 (括弧囲み，強調)
$E_r(w, w_a)$:注目語に対する重要度	$F_c(w, w_a)$:共起 (注目語と共起する頻度) $F_d(w, w_a)$:距離 (注目語との間の文字数)

w_a:注目語　　w:周辺語

式 (4.2) の $F_m(w)$，$F_v(w)$，$p(w)$ は，それぞれ式 (4.3)，式 (4.4)，式 (4.5) で定義する．ここで，D_M は形態素解析辞書が持つ名詞種別の辞書を表し，D_B は括弧 (「 」，( )，“ ”, 【 】, 『 』，など) および HTML ソースのタグによる強調表現 (⟨b⟩，など) を含む，言語にほとんど依存しない小規模な辞書を表す．$df(w)$ は w が出現するドキュメント数を表し，N はそのドキュメント集合に含まれる全ドキュメント数を表す．

$$F_m(w) = \begin{cases} 1 & (w \in D_M) \\ 0 & (w \notin D_M) \end{cases} \tag{4.3}$$

$$F_v(w) = \begin{cases} 1 & (w \in D_B) \\ 0 & (w \notin D_B) \end{cases} \tag{4.4}$$

$$p(w) = \log_2 \frac{N}{df(w)} \tag{4.5}$$

(2)$E_r(w, w_a)$ の算出

共起語は注目語からの距離が離れるほど関連する度合いが小さくなると仮定する．$E_r(w, w_a)$ は，$F_c(w, w_a)$ を $F_d(w, w_a)$ で割った値とし，式 (4.6) に従い算出する．

$$E_r(w, w_a) = \frac{F_c(w, w_a)}{F_d(w, w_a)} \tag{4.6}$$

式 (4.6) の $F_c(w, w_a)$ と $F_d(w, w_a)$ は，それぞれ式 (4.7) と式 (4.8) で定義する．$freq(w)$ は w の出現数を，Doc は表示中の Web ページ中のテキスト全体を表す．また $dist(w, w_a)$ は，w と w_a の間にある文字数に基づく表示上の距離を表す．k は w に対して共起する w_a の候補を選択する範囲とする．

$$F_c(w, w_a) = \frac{freq(w|dist(w, w_a) < k)}{freq(w|w \in Doc)} \tag{4.7}$$

$$F_d(w, w_a) = dist(w, w_a) \tag{4.8}$$

4.5 性能評価

提案方式の有効性を評価するため，実際の Web ページを利用した評価実験を行った．4.4 節で述べた要素に基づく提案方式を実装したシステムを構築し，従来方式と提案方式それぞれについて 3 人の被験者による主観評価実験を行った．なお，本節での評価の目的は周辺語を利用した Web ページ推薦の精度を測ることであることから，リモコン型 UI の形状や機能，方式などの評価への影響を避けるため，ポインティングデバイスとして一般的なマウスを利用した．また，クリック位置がずれてしまい，意図しない語を注目語としてしまうなどのクリック失敗の度合いについてもリモコン型 UI の種類に依存するため，本評価では検証しない．

まず，評価用の注目語を収集するための実際の Web ページとして，表 4.2 に示す 5 つのカテゴリに属する Web ページをそれぞれ 25 ページずつ，合計 125 ページを収集した．ここから，3 人の被験者にそれぞれ各ページ毎に 2 つの注目語を選択してもらい，合計 750 個の評価用の注目語を収集した．

以下，4.5.1 項では予備実験として，提案方式で良い検索結果が得られるための最適な検索語数を検証し，さらに，提案方式を適用することで検索結果の満足度を改善可能な場合がどの程度存在するかを検証する．次に，4.5.2 項では，4.5.1 項 (2) で得られた注目語のみでは十分に満足する検索結果が得られない場合において，提案方式を適用することでどの程度の満足度の改善が見込めるかを検証する．ここでは，検索エンジンとして Yahoo!API[67] を利用した．

表 4.2: 評価対象のカテゴリ

カテゴリ名	概要
IT 系	IT 関連の情報を提供している Web ページ
メディア系	テレビ局などが提供している Web ページ
医療健康系	医療・健康関連の情報を提供している Web ページ
ニュース系	新聞社が提供しているニュースの Web ページ
京都観光系	京都の観光情報を提供している Web ページ

4.5.1 予備実験

(1) 提案方式に最適な検索語数の検証

- 実験方法
  検索語数を 1〜5 個に調整して検索した結果の第 1 位の Web ページの満足度を評価した．検索語の選択方法は，提案方式のパラメタの影響を避けるため，最も直観的な指標である注目語からの距離 ($F_d(w, w_a)$) に基づき選択した．検索語数が 1 個の場合は，注目語のみでの検索を意味し，2 個目以降の検索語は注目語からの距離が近い順に選択した．評価値は 3 段階 (最低 0 点，最高 2 点) とし，それぞれ,「注目語とまったく見当違いの内容，または検索結果が 0 件 (0 点)」,「注目語の意味と内容がやや関連する (1 点)」,「注目語の意味と内容がぴったり合っている (2 点)」を評価基準とした．

- 結果と考察
  カテゴリ毎に 3 人の評価値を平均した結果，および全カテゴリの結果の平均値を図 4.3 に示す．平均値は 3 語以上でほぼ安定していることから，本論文における提案方式の検索語数は，注目語を含めて 3 個とする．

(2) 提案方式を適用する範囲の検証

- 実験方法
  注目語のみで検索した結果の第 1 位の Web ページの満足度を評価した．評価値は 5 段階 (最低 0 点〜最高 4 点) とし，それぞれ,「検索意図とまったく見当違いのページ内容，または検索結果が 0 件 (0 点)」,「検索意図とページ内容が部分的に合っている

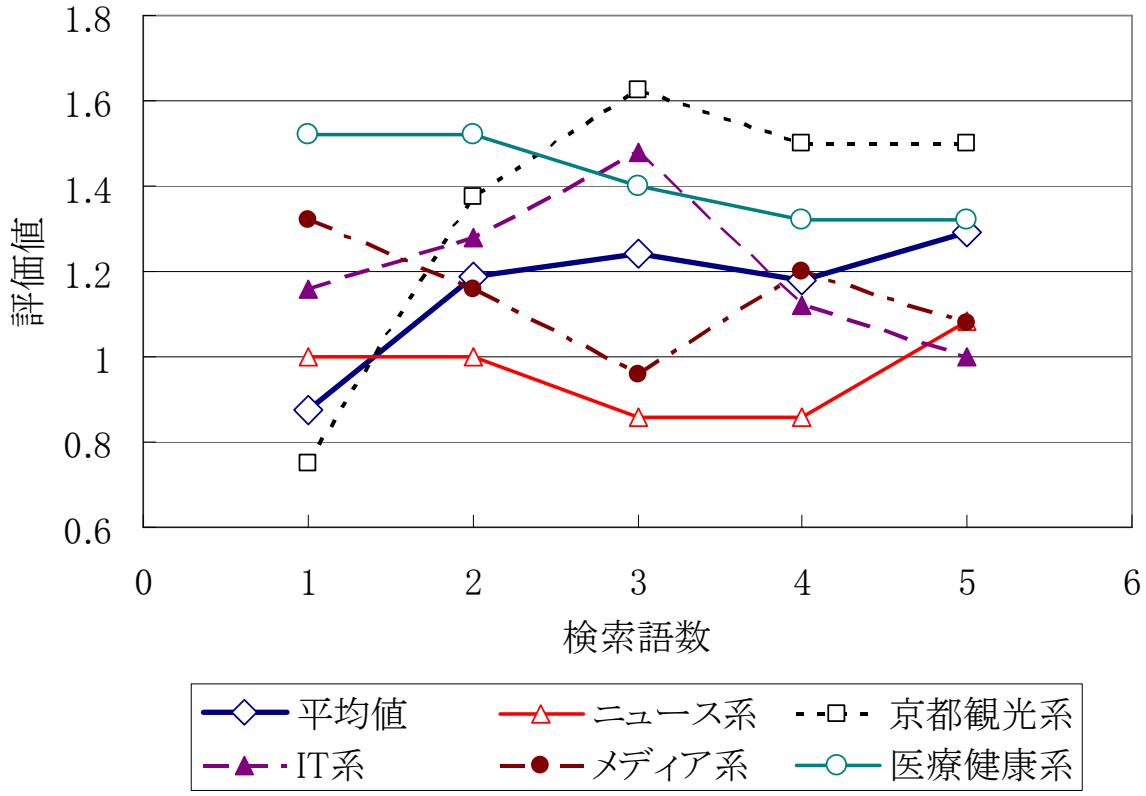


図 4.3: 検索語数と評価値

(1 点)」,「検索意図とページ内容の主題が合っている (2 点)」,「検索意図に対してページ内容に満足できるレベルの記述がある (3 点)」,「検索意図とページ内容がぴったり合っている (4 点)」とし，4 点は目的の情報が確実に推薦されたと思った場合にのみ付けることを条件とした．このようにして得られた評価結果を，注目語のみの検索結果で 3 点以上の評価となったページと，2 点以下の評価となったページに分類した．2 点以下のページは多くのユーザが欲するような標準的な情報では満足しなかった検索結果を表しており，提案方式による改善が見込まれる範囲を表している．

- 結果と考察

  結果を表 4.3 に示す．この結果より，カテゴリにより多少のばらつきがあるものの，平均すると 50%以上の場合において，注目語のみでは検索結果に満足していない結果となった．また，IT 系と京都観光系を比較すると大きく結果が異なった．この要因として，IT 系の語は,「オンラインチャージ」「オープンシステム」「フレームワー

表 4.3: 注目語のみによる検索の評価結果 (括弧内の数値はサンプル数 150 件に対する割合を表す)

	3 点以上 (注目語のみの検索で満足)	2 点以下 (提案方式で改善が見込まれる範囲)
IT 系	51(34.0%)	99(66.0%)
メディア系	64(42.7%)	86(57.3%)
医療健康系	70(46.7%)	80(53.3%)
ニュース系	70(46.7%)	80(53.3%)
京都観光系	95(63.3%)	45(36.7%)
平均	70(46.7%)	80(53.3%)

ク」などのように，使われる範囲が限定されないと意味を限定しにくい語が多いことや,「ファイアウォール」などのようにユーザの知りたい範囲により内容や記述の難易度が大きく異なる語が多いこと，さらに，それらの語が多く出現する Web ページであってもその説明がなく使われている場合が多いことから，検索結果に対するユーザの検索意図との乖離が起こりやすいと考えられる．一方，京都観光系の語は著名な寺社仏閣などの名所名や歴史上の人物名に関する語が多く，その語を説明する内容は史実に基づく不変の内容がほとんどであるため，検索結果に対するユーザの検索意図との乖離が起こりにくいと考えられる．

4.5.2 提案方式の評価

- 実験方法

  提案方式と，注目語のみで検索した場合 (以降，従来方式と呼ぶ) の検索結果の満足度を比較するため，3 名の評価者に対して 4.5.1 項 (2) と同様に 5 段階評価による主観評価実験を行った．まず従来方式の結果の評価を行い，続いて提案方式の結果の評価を行った．

  従来方式の結果に対して提案方式の満足度が高い場合 (“評価が上がった”)，下がった場合 (“評価が下がった”)，変化しなかった場合 (“変化なし”) をそれぞれカウントし

た．3 人の被験者の評価を総計した結果を表 4.4 に示す．また，従来方式で 3 点以上の評価となった場合 (“注目語のみで満足できた”) に対しても同様に，提案方式の満足度の変化をカウントした．結果を表 4.5 に示す．α の値は，学習データとして任意の 10 サイトをサンプルとして抽出し，式 (4.1) の第 1 項の平均値と第 2 項の平均値が同じになるように算出した ($\alpha = 0.06081$)．また，式 (4.7) の k の値は，周辺語が 3 語〜5 語抽出できる程度になるように実験的に決定し，$k = 20$ とした．ここで，式 (4.5) の $df(w)$ の算出に必要なドキュメント集合として，1994 年から 2006 年までの毎日新聞オンラインニュースの 135 万件の記事を利用した．

- 結果

  表 4.4 より，従来方式で 2 点以下の場合においては，医療健康系および京都観光系のカテゴリにおいて提案方式が従来方式を大幅に上回る評価結果が得られた．特に，京都観光系のカテゴリにおいて，63.6%のケースで有効性が確認できた．その他の 3 つのカテゴリについても提案方式が上回っており，提案方式の有効性を確認できた．また，表 4.5 より，従来方式で 3 点以上の場合においては，京都観光系のカテゴリにおいて評価結果が均衡しているものの，すべてのカテゴリを総合的に見ると，71.5%のケースにおいて提案方式が注目語のみの検索結果を下回った．よって，従来方式の検索結果にユーザが満足できるケースにおいては，提案方式が上回る可能性が低いと言える．

表 4.4: 注目語のみで検索する方式の評価が 2 点以下の場合における提案方式の評価との比較 (カッコ内の数値は「評価が上がった」「変化なし」「評価が下がった」の合計値に対する割合を示す)

	評価が上がった	変化なし	評価が下がった
IT 系	29(29.3%)	55(55.6%)	15(15.1%)
メディア系	18(21.0%)	53(61.6%)	15(17.4%)
医療健康系	29(36.3%)	49(61.2%)	2(2.5%)
ニュース系	30(37.5%)	41(51.3%)	9(11.2%)
京都観光系	35(63.6%)	20(36.4%)	0(0%)
合計	141(35.3%)	218(54.5%)	41(10.2%)

表 4.5: 注目語のみで検索する方式の評価が 3 点以上の場合における提案方式の評価との比較 (括弧内の数値は「評価が上がった」「変化なし」「評価が下がった」の合計値に対する割合を表す)

	評価が上がった	変化なし	評価が下がった
IT 系	0(0%)	13(25.5%)	38(74.5%)
メディア系	0(0%)	7(10.9%)	57(89.1%)
医療健康系	0(0%)	21(30.0%)	49(70.0%)
ニュース系	1(1.4%)	12(17.1%)	57(81.5%)
京都観光系	4(4.2%)	42(44.2%)	49(51.6%)
合計	5(1.4%)	95(27.1%)	250(71.5%)

- 考察

  以上の結果より，提案方式と従来方式は単独では十分な満足度を実現できていないといえる．ここで，表 4.4 で示したように従来方式で満足できない場合は提案方式で改善できること，および表 4.5 で示したように従来方式で満足しているケースにおいては，提案方式を適用しても効果が薄いことが観測できる．このことから，まず従来方式の結果をユーザに提示し，その結果満足しないリアクションが得られた場合には続いて提案方式の結果を提示するなど，提案方式と従来方式を組み合わせた実装方法が有効であると考えられる．これについては 4.5.4 項のシステム実装例で検証する．

  また，個々の評価結果を観測すると，提案方式が有効に作用するケースを次の 3 つに分類できる．

  ケース 1：注目語だけでは文章量が少なく，十分な情報が得られないケース
  ケース 2：注目語だけでは法律の原文などの資料そのものが検索されてしまうケース
  ケース 3：注目語の部分一致で検索され，注目語そのものを検索できないケース

  それぞれのケースについて，提案方式が有効に作用した具体例を表 4.6 から表 4.8 に示す．

表 4.6: 検索結果の具体例ケース 1(太字は抽出した注目語を表す)

		従来方式	提案方式
例 1-1	原文	「· · · 今回は明治から昭和初期に活躍した「**神坂雪佳**」を紹介する．当時多くの芸術家が西欧へ · · ·」	
	検索語	神坂雪佳	神坂雪佳，明治，昭和初期
	結果ページ概要	文章量が比較的少ないページ	文章量が比較的多いページ
	ユーザ評価値	2	4
例 1-2	原文	「· · · 質調査所によると，インドネシア・**ジャワ島沖**で九日午前零時（日本時間同二時）ごろ，· · ·」	
	検索語	ジャワ島沖	ジャワ島沖，九日午前零時，日本時間
	結果ページ概要	要点だけを述べた比較的短いニュース記事	文章量が比較的多い詳しいニュース記事
	ユーザ評価値	1	2
例 1-3	原文	「· · · 尾，京の吉野と並び称された大阪の名妓「**夕霧**」の墓がある清涼寺で追善法要が行われま · · ·」	
	検索語	夕霧	夕霧，京，吉野
	結果ページ概要	文章量が比較的少ないページ	文章量が比較的多いページ
	ユーザ評価値	1	2
例 1-4	原文	「· · · １８５５年) に造営されたものです．**紫宸殿**を始めとし，清涼殿，小御所，御学問所及び · · ·」	
	検索語	紫宸殿	紫宸殿 学問所 清涼殿
	結果ページ概要	文章量と写真の枚数が比較的少ないページ	文章量が多く，写真を多用したページ
	ユーザ評価値	3	4

一方，提案方式が有効に作用しない事例として，関連する文書の種類が少ない場合において，周辺語を含む Web ページとして実験対象の Web ページが推薦されてしまうケースがあった．また，4.4 節で述べた提案方式の (Step2) で生成した複合語が，検索サイトの検索語として登録されていないものであったため，検索結果が 0 件となってしまったケースが見られた．前者については，URL の一致を見ることで解決可能で

表 4.7: 検索結果の具体例ケース 2(太字は抽出した注目語を表す)

		従来方式	提案方式
例 2-1	原文	「･･･財政改革や格差問題などが争点.「**従軍慰安婦**」などをめぐる上田氏の歴史認識も議論と･･･」	
	検索語	従軍慰安婦	従軍慰安婦，上田氏，格差問題
	結果ページ概要	このタイトルの本の目次と感想が書かれたページ	この問題に関連するニュース記事
	ユーザ評価値	1	2
例 2-2	原文	「･･･が米艦船などに給油活動するための**テロ対策特別措置法**が１１月１日で期限切れとなるこ･･･」	
	検索語	テロ対策特別措置法	テロ対策特別措置法，米艦船，給油活動
	結果ページ概要	法律の原文がそのまま記載されたページ	法律に詳しくない人でも分かるように平易に法律の内容を説明しているページ
	ユーザ評価値	3	3(評価値は変わらず)
例 2-3	原文	「･･･が揺らいでいないか，と突き付ける内容の**平和宣言**を読み上げた．銃撃され死亡した伊藤･･･」	
	検索語	平和宣言	平和宣言，内容，銃撃
	結果ページ概要	宣言の原文がそのまま記載されたページ	原文を引用し事件にも触れて著者の意見も含むページ
	ユーザ評価値	2	3

あり，後者については，検索結果が 0 件となるとき，複合語を生成前の状態に戻して再検索をすることで解決可能であると考えられる．また，ケース 1 とケース 2 の具体例では，注目語に対する情報量が多いページが提示できたために提案方式が高い評価を得ている．表 4.4 のジャンル間の結果を参照すると，観光名所の歴史を知りたい場合や医療に関する参考情報を知りたいなど，注目語に対する関連情報を検索したい場合には提案手法が有効であり，一方，ニュースなどに出現する単語の意味を知りたい場合は，注目語のみで検索する方が有効であると考えられる．つまり，ユーザが検索結果として求める情報の違いがジャンルに依存している可能性が高いと考えられる．

表 4.8: 検索結果の具体例ケース 3(太字は抽出した注目語を表す)

		従来方式	提案方式
例 3-1	原文	「···爆発し，乗組員 7 人全員が死亡した.「**宇宙の先生**」計画は中断され，モーガンさんは学···」	
	検索語	宇宙の先生	宇宙の先生，乗組員 7 人全員，モーガンさん
	結果ページ概要	「宇宙」を含むページ	事故に関連するニュースのページ
	ユーザ評価値	1	2

4.5.3 各要素の影響度評価

提案した周辺語重要度モデルの個々の要素について評価実験を行い，提案方式の有効性に寄与した要因を検証する.

(ア) $F_m(w)$ および $F_v(w)$ の評価

- 実験方法

  $F_m(w)$ 要素と $F_v(w)$ 要素の有効性の評価を行った．$F_m(w)$ の評価のための検索語として，注目語に加えて，形態素解析で「名詞–人名」「名詞–地名」と判定された周辺語を 2 語選択した．不足する場合は，注目語からの距離が近い語から順に選択した．注目語の近隣に人名および地名を含むという条件に該当する評価結果について，検索結果第 1 位の Web ページに対する評価値を集計した．一方，$F_v(w)$ の評価のための検索語として，出現数の多い「 」,『 』，および タグで囲まれた語を 2 語選択した．不足する場合は同様に，注目語からの距離が近い語から順に選択した．注目語の近隣に強調表現を含むという条件に該当する評価結果についても同様の評価値を集計した．なお，これらの条件に該当する評価結果の数が少ないため，ここでは 5 つのカテゴリの結果を総計し，地名優先の評価総数は 53，人名優先の評価総数は 29，強調表現優先の評価総数は 154 となっている．4.5.2 項と同様の手法で従来方式に対する評価結果の差を集計した.

- 結果と考察

  結果を表 4.9 に示す．この結果より，周辺語に地名を優先的に利用することで，評価

表 4.9: 人名優先と地名優先および強調表現優先の評価結果

	評価が上がった	変化なし	評価が下がった
地名優先	12	36	5
人名優先	4	19	6
強調表現優先	55	66	33

が上がっており，地名を優先的に利用する効果は高いといえる．一方，人名を優先的に利用した場合は，ほとんど評価が変わっていない．よって，地名を利用する場合と比較して人名を利用する場合の効果は薄いといえる．また，周辺語に強調表現の語を利用することで評価が上がっており，強調表現の語を優先的に利用する効果は高いといえる．

(イ) $F_c(w, w_a)/F_d(w, w_a)$ の評価

- 実験方法
  $F_c(w, w_a)$ 要素と $F_d(w, w_a)$ 要素の有効性の評価を行った．検索語は注目語に加えて，$F_r(w, w_a)$ の値が大きい順に周辺語を 2 語選択した．従来方式との評価結果との差について，4.5.2 項と同様の評価手順により集計した．

- 結果と考察
  結果を表 4.10 に示す．この結果より，医療健康系および京都観光系サイトでは，従来方式を大幅に上回る結果となっており，従来方式で満足できなかった場合は，提案方式を利用することでほとんどの場合は満足度が上がっている．またその他 3 カテゴリについても，多くの場合，満足度が向上している．以上により，共起語を検索語に加える優先度を上げることは，検索結果の満足度向上に有効であるといえる．

4.5.4 システム実装例

提案方式を実装した検索システムを構築した．機能構成と動作シーケンスを図 4.4 に示す．本システムは，クリック検索サーバを中心として，Web ブラウザをユーザインタフェースとした端末，一般の Web サイト，および一般の検索エンジンが提供している検索 API サー

表 4.10: 共起語優先の評価結果

	評価が上がった	変化なし	評価が下がった	注目語のみで満足できた
IT 系	38	41	20	51
メディア系	33	42	10	65
医療健康系	42	34	4	70
ニュース系	23	35	11	81
京都観光系	34	37	0	79
合計	170	189	45	346

バの 4 つの機能から構成される．ここで，ユーザがクリックした位置の前後の文字列を取得する機能を Web ブラウザのプラグインとして実装した．また，4.5.2 項の考察に従い，注目語のみの検索で得られた第 1 位の Web ページをユーザに第 1 番目に提示する Web ページとし，第 2 番目以降の検索結果として提案方式の検索結果を続けるように実装した．

さらに，クリック検索サーバが応答した Web ページの良否を評価する機能をクリック検索サーバに実装し，その評価用 GUI を端末の Web ブラウザのプラグインとして実装した．本評価機能を利用して，注目語に対するユーザが良いと判断した Web ページをクリック検索サーバに蓄積することができるようになる．この評価情報を利用することで，ソーシャルフィルタとして検索結果に反映する機能が利用できる．この機能は，評価情報がサービス開始時などの初期状態では存在しないことや，時間の経過に伴う Web ページ自体の情報更新などに対応していないという問題が残されている．この点については，今後の課題と考えている．システムの処理は，図 4.4 に記載している処理シーケンス (1)〜(16) の手順で行う．

以下に各処理シーケンスの概要を述べる．

(1) クリック操作

ユーザが，Web ブラウザ上の任意の語をポインタでクリックする．

(2) 文字列送信

端末の Web ブラウザが，ポインタ位置の前後の文字列を切り出してクリック検索サーバに送信する．

(3) 検索語生成

クリック検索サーバが，端末から受信した文字列を利用して注目語と周辺語を抽出し，周辺語重要度に基づき検索語を生成する．このとき，注目語に対する十分に信頼できる評価情報が蓄積されている場合は，その Web サイトにページを要求し，(8) へ進む．

(4) 検索要求

クリック検索サーバが，検索 API に対して検索語を入力して検索結果を要求する．このとき，(a) 注目語のみ，(b) 注目語と周辺語の組み合わせの 2 通りで要求する．

(5) 検索応答

検索 API が，(a)(b)2 つの検索結果を応答する．

(6) 検索結果構築

クリック検索サーバが，2 つの検索結果を合成した検索結果リストを構築する．具体的には，(a) の検索結果の第 1 位を 1 番目とし，2 番目以降に (b) の結果をつなぐ．

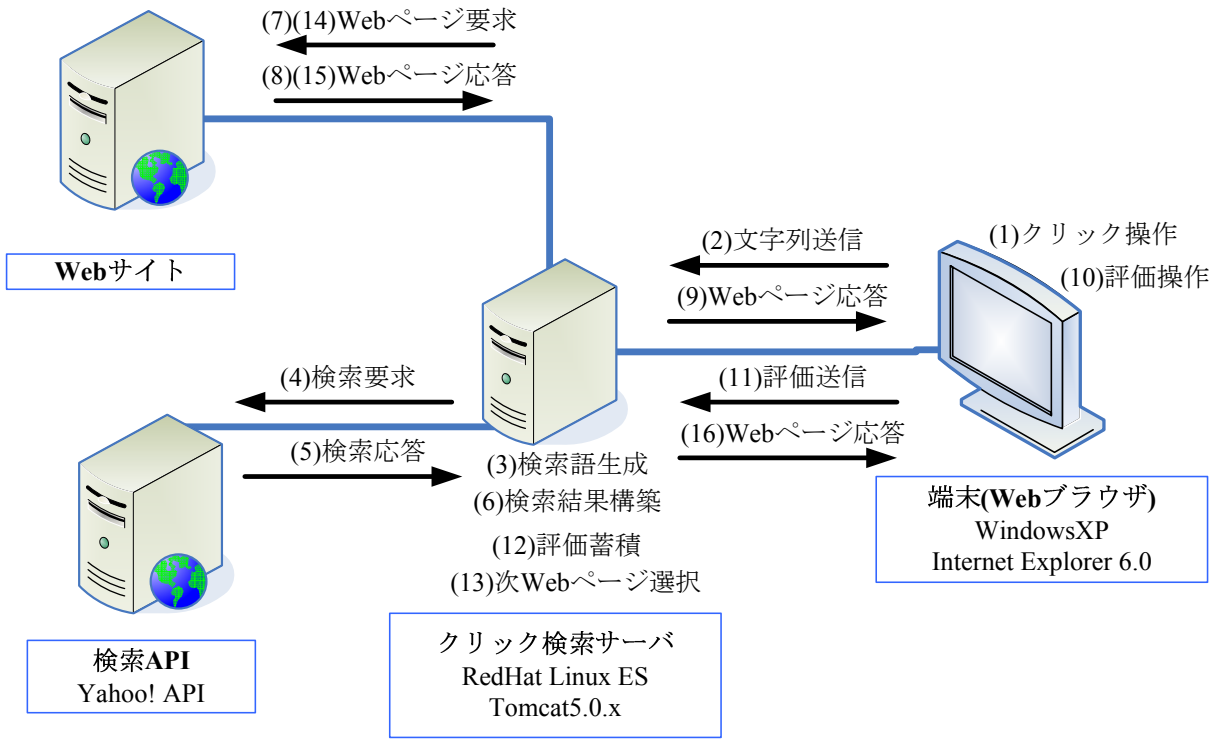


図 4.4: クリック検索システムの実装例

(7) Web ページ要求

クリック検索サーバが，検索結果リストの 1 番目の URL に基づきページを Web サイトに要求する．

(8) Web ページ応答

Web サーバが，要求された Web ページをクリック検索サーバに応答する．

(9) Web ページ応答

クリック検索サーバが，検索結果として 1 番目の Web ページを端末に応答する．

(10) 評価操作

ユーザが，検索結果の Web ページを閲覧し，検索結果として満足する情報を含む Web ページかどうかを評価する．満足した場合は，閲覧は終了する．満足しない情報であった場合は，低い評価を付与することで，次の Web ページを要求する．

(11) 評価送信

端末の Web ブラウザが，ユーザの評価結果をクリック検索サーバに送信する．

(12) 評価蓄積

クリック検索サーバが，ユーザの評価情報を，提示した Web ページと紐付けて蓄積する．

(13) 次 Web ページ選択

ユーザの評価が低い評価であった場合，クリック検索サーバが，検索結果リストの次の Web ページを選択する．

(14) Web ページ要求

クリック検索サーバが，次の Web ページの URL に基づき，ページを Web サイトに要求する．

(15) Web ページ応答

Web サーバが，要求された Web ページをクリック検索サーバに応答する．

(16) Web ページ応答

クリック検索サーバが，検索結果として次の Web ページを端末に応答する．

以降，(10)〜(16) を繰り返す．

以上に示した実装方法と，従来方式の検索結果の順に表示した場合の満足度の違いについて評価した．第 1 番目に提示した Web ページの評価が 2 以下だった場合に，2 番目以降，5 番目以内に提示される Web ページの中で，評価が 3 以上の Web ページをどちらが早く提

示できたかを比較した．この結果を表 4.11 に示す．この表では，提案方式が先に提示できた場合を「提案が優位」，同順だった場合を「同順」，従来方式が先に提示できた場合を「従来が優位」として，それぞれのカテゴリについて集計している．

この結果から，医療健康系および京都観光系のカテゴリにおいて，提案方式が有効であることが分かる．これら 2 つのカテゴリについては，表 4.4 に示した提案方式の評価結果において顕著な有効性が見られたカテゴリであり，この評価実験により 1 位以外の検索結果においても提案方式が有効であることを示した．他の 3 つのカテゴリについては，表 4.4 においては若干の有効性は得られたものの，この評価実験により，1 位以外の検索結果においては，有効に作用しない結果となった．この理由として，提案方式では多くの検索語を用いているため検索結果の内容にバリエーションがなく，結果が外れた場合の補完ができる可能性が低いことが挙げられる．そのため，比較的結果が外れる可能性の高い 3 つのカテゴリにおいて，有効に作用しない結果となったと考えられる．これについては，提案方式と従来方式の検索結果を内容に応じて適応的に混合するなど，検索結果の提示方法の検討が必要である．

4.6 むすび

本章では，入力インタフェースとして広く利用されているリモコン型 UI を用いた Web 検索操作の簡略化を目標とし，クリック型検索手法を提案した．本手法は，ユーザがポインタで任意に指定した注目語について，システムが自動的に周辺語を補完して検索を行い，検索結果をユーザに提示することができる．

表 4.11: 実装例における検索結果の従来方式との比較評価

	提案が優位	同　順	従来が優位
IT 系	19	44	36
メディア系	10	42	33
医療健康系	25	33	22
ニュース系	8	50	21
京都観光系	23	22	10

周辺語の重要度を決定するための周辺語重要度モデルについて，語自身の重要度と注目語との関連性の 2 点からなるモデルを提案し，検索結果に対する満足度を測定するための主観評価実験を行った．その結果，注目語のみを検索語とした検索結果では 50%以上の場合で満足しないことがわかり，そのうち，提案方式を適用することで最大で 63.6%の検索クエリで満足度が向上することを示した．さらに，周辺語重要度を算出するための個々の要素について評価を行い，地名と強調文字，および注目語に対する共起語を優先的に検索語に利用することで，検索結果に対する満足度が向上することを示した．ただし，ジャンルにより結果は異なり，提案方式が得意なケースとそうでないケースに分かれた．適切な検索結果が，注目語に関連する量の多い情報なのか，単語の意味解説などの特定の情報なのかなど，どのような種類の情報をユーザが検索結果として求めているのかが評価結果に影響している可能性が高い．

実用化に向けた今後の課題として，ユーザが検索結果として求める情報の種類と評価結果の関連性について検証するなど，ジャンルと検索結果の評価の関連性を明確にする必要がある．また，その結果に基づき，ジャンルに応じて，あるいはユーザが求める情報の種類に応じて周辺語の組み合わせ方を適応的に変更する手法の検討が必要である．

第5章

結論

本論文では，PC 向けの Web ページの情報を小型情報端末で検索・閲覧する作業を支援する情報収集支援手法について検討した．PC だけでなく携帯電話やゲーム機，カーナビゲーションシステム，STB(Set Top Box) などのさまざまな小型情報端末が Web ブラウザと通信機能を搭載するようになってきたことから，キーボードのない環境においても検索・閲覧できるようにする技術の確立は非常に重要かつ広範な課題である．

そこで本論文では，この課題を解決するためのアプローチとして，まず第 1 章で，課題を 4 つの技術領域に分類して整理した．次に，分類した領域のうちの 3 つの領域において，従来技術では実現できていない課題を抽出した．3 つの課題とは，(1) 特定の情報を時系列観測する作業におけるユーザの負荷を減らすこと，(2) 低解像度画面の端末で効率的に情報閲覧できるようにすること，(3) キーボードが使えない端末で検索操作を容易に行えるようにすること，である．それぞれの課題は，第 2 章，第 3 章，第 4 章で順に解決した．

第 2 章では，特定の情報を時系列観測する作業におけるユーザの負荷を減らすことを目的として，小型情報端末向けの Web からの時系列情報自動抽出手法を提案した．本手法により，ユーザが着目した任意の情報に対し，その Web ページ内で識別可能な ID を自動設定することで，ランキングが変動するなどにより Web ページ内の表示位置やその値が変わった場合でも，自動追跡が可能となった．1 行または 1 列でオブジェクトが表現されている一般的な表形式情報に提案手法を適用して評価した結果，ユーザは情報を指定する操作を一度行うだけで，時間変化する情報を高い確率で追跡可能となることを示した．これにより，小型情報端末における情報収集作業の大幅な省力化を実現した．

第 3 章では，低解像度画面の端末で効率的に情報閲覧できるようにすることを目的とし

て，小型情報端末向けのWebページ分割手法を提案した．本手法により，ユーザが指定した任意のWebページに対し，タグの深さに基づくタグ構造情報とテーブル表現などを利用したレイアウト情報を利用して，ページ内の情報の意味的なつながりを高精度に推定し，複数の小さなWebページに自動分割することが可能となった．ユーザは，分割された小さなWebページのリストから好みの情報を選択する簡単な操作で，目的の情報を閲覧可能となり，小型情報端末における情報閲覧の大幅な操作数の削減を実現した．

第4章では，キーボードが使えない端末で検索操作を容易に行えるようにすることを目的として，小型情報端末向けのワンクリック検索手法を提案した．本手法により，表示中のWebページ内にユーザが調べたい注目語があった場合において，注目語の周辺から注目語を修飾する重要な語を抽出することで，検索キーワードの自動生成が可能となり，容易に検索結果を得ることを可能にした．ユーザは，気になる語をクリックするだけで，検索結果を得ることが可能となり，小型情報端末における情報検索操作の簡略化を実現した．評価実験を行い，検索結果の満足度はジャンルによって異なり，提案手法が得意なケースとそうでないケースに分かれることがわかり，ジャンル等に応じて適応的に検索語を変更する必要があることを明確にした．

本論文の成果により，これまで，高度の入出力インタフェースを持つPCのような高機能な端末でなければ困難であった，Web上の情報を検索と閲覧を繰り返して情報収集する操作を，小型情報端末上でも容易に行えるようになった．これにより，いつでもどこでもWeb上の情報を容易に活用できるようになり，小型情報端末が人の情報処理能力や行動を本格的に支援するツールとなるための，1つのステップを解決したといえる．

今後の課題として，Webの表層的な特徴だけでなく，HTMLソースのタグの意味やテキストの内容等の意味的な特徴を抽出して利用することで，適用範囲や精度の向上を目指した改善が挙げられる．まず，小型情報端末向け時系列Web情報自動抽出・提示方式においては，現在のアルゴリズムでは全ての表形式情報には対応できないため，任意の表形式の情報についても対応できる手法について検討し，適用範囲を広げる必要がある．ここでは，表形式情報の表層的な特徴を利用してオブジェクトを抽出しているが，HTMLソースやテキストの内容など，意味的な特徴を利用して新たなオブジェクトの概念を導入するなど，新たな方式を検討する予定である．次に，小型情報端末向けWebページ自動分割手法においては，分割精度向上のための改善が挙げられる．ここでは，HTMLソースのタグ構造や，表などの一部のタグの意味を理解して処理を行っているが，さらにその他のタグの意味や

スタイル情報，画像の意味などの，より深い意味を抽出して利用するなど，新たな方式を検討し，導入する必要がある．また，小型情報端末向けクリック型 Web 検索手法においては，周辺語重要度のモデルの各要素の精細化や要素の追加，ならびに複数の注目語を入力する等の機能拡張により，検索結果に対するユーザの満足度の更なる向上を目指すことが挙げられる．また，技術の実用化に向けて，実際のリモコン型 UI を本システムに適用し，操作数などの評価実験を行う必要がある．さらに，提示する Web ページの満足度向上のため，検索結果の提示方法についても検討する必要がある．

謝辞

本研究全般に関して，懇切なる御指導と惜しみない御助言を頂きました大阪大学大学院情報科学研究科マルチメディア工学専攻 西尾章治郎教授に謹んで御礼申し上げます.

本研究を推進するにあたり，直接の御指導，御助言，御討論を頂きました大阪大学大学院情報科学研究科マルチメディア工学専攻 原隆浩准教授に心より感謝申し上げます.

本論文をまとめるにあたり，大変有益な御指導と御助言を多数頂きました大阪大学大学院情報科学研究科マルチメディア工学専攻 岸野文郎教授，鷹田憲久教授に心より感謝申し上げます.

本論文をまとめるにあたり，御指導と御助言を頂きました大阪大学大学院情報科学研究科マルチメディア工学専攻 藤原融教授，大阪大学サイバーメディアセンター 下條真司教授に心より感謝申し上げます.

本論文をまとめるにあたり，御指導を頂きました大阪大学大学院情報科学研究科マルチメディア工学専攻 寺西裕一准教授，大阪大学サイバーメディアセンター 義久智樹講師，大阪大学大学院情報科学研究科マルチメディア工学専攻 神崎映光助教に心より感謝申し上げます.

本研究を行うにあたり，貴重な御討論を頂きました大阪大学大学院情報科学研究科マルチメディア工学専攻 荒瀬由紀氏，大西健史氏，小牧大治郎氏，大阪大学工学部電子情報工学科 岩田麻佑氏に深く御礼申し上げます.

本研究を進めるにあたり，多くのご支援を頂きました大阪大学大学院情報科学研究科マルチメディア工学専攻 西尾研究室の諸氏に深く御礼申し上げます.

本研究を進める上で，惜しみない御助言，御協力，多大なる御支援を頂きました株式会社 KDDI 研究所 所長 秋葉重幸博士，副所長 松本修一博士，東京大学大学院情報理工学系研究科 浅見徹教授に心より感謝申し上げます.

本研究をともに進め，有益な御指導，御助言，御討論，および研究活動を進めるにあたっ

てのご支援を頂きました株式会社国際電気通信基礎技術研究所 適応コミュニケーション研究所 所長 小花貞夫氏，独立行政法人 情報通信研究機構 西山智氏，株式会社 KDDI 研究所 執行役員 菅谷史昭氏，執行役員 堀内浩規氏，グループリーダー 滝嶋康弘氏，主任研究員 松本一則氏，グループリーダー 杉山敬三氏，主任研究員 小野智弘氏，主任研究員 吉原貴仁氏，研究主査 帆足啓一郎氏，研究主査 上向俊晃氏，研究員 武吉朋也氏に心より感謝申し上げます.

最後に，大学院において研究生活を送る上で，多大なる御支援と御理解を頂いた両親と家族に心からの感謝と御礼を申し上げます.

参考文献

[1] C. R. Anderson, P. Domingos, and D. S. Weld: Personalizing Web Sites for Mobile Users, in *Proceedings of International World Wide Web Conference* (*WWW 2001*), pp. 565–575 (May 2001).

[2] J. L. Arjona, R. Corchuelo, D. Ruiz, J. Peña, and M. Toro: Knowledge Channels: Bringing the Knowledge on the Web to Software Agents, in *Proceedings of Conference on Advanced Information Systems Engineering*, *http://SunSITE.Informatik.RWTH-Aachen.de/Publications/CEUR-WS//Vol-74/./files/FORUM_41.pdf* (June 2003).

[3] J. L. Arjona, R. Corchuelo, and M. Toro: A Knowledge Extraction Process Specification for Today's Non-Semantic Web, in *Proceedings of Web Intelligence* (*WI 2003*), p. 61 (Oct. 2003).

[4] Alexa, *http://www.alexa.com/*.

[5] au one ポータル, *http://auone.jp/*.

[6] au one ラボ, *https://labs.auone.jp/*.

[7] P. Baudisch, X. Xie, C. Wang, and W. Ma: Collapse-to-Zoom: Viewing Web Pages on Small Screen Devices by Interactively Removing Irrelevant Content, in *Proceedings of Symposium on User Interface Software and Technology* (*UIST 2004*), pp.91–94 (Oct. 2004).

[8] G. Buchanan, S. Farrant, M. Jones, and H. Thimbleby: Improving Mobile Internet Usability, in *Proceedings of International World Wide Web Conference*(*WWW 2001*), pp. 673–680 (May 2001).

[9] O. Buyukkokten, H. Garcia-Molina, A. Paepcke, and T. Winograd: Power Browser: Efficient Web Browsing for PDAs, in *Proceedings of Human-Computer Interaction Conference* (*CHI 2000*), pp. 430–437 (Apr. 2000).

[10] O. Buyukkokten, H. Garcia-Molina, and A. Paepcke: Seeing the Whole in Parts: Text Summarization for Web Browsing on Handheld Devices, in *Proceedings of International World Wide Web Conference* (*WWW 2001*), pp. 652–662 (May 2001).

[11] Y. Chen, W. Y. Ma, and H. J. Zhang: Improving Web Browsing on Small Devices Based on Table Classification, in *Proceedings of International World Wide Web Conference* (*WWW 2003*), pp. 20–24 (May 2003).

[12] Y. Chen, W. Ma, and H. Zhang: Detecting Web Page Structure for Adaptive Viewing on Small Form Factor Devices, in *Proceedings of World Wide Web Conference* (*WWW 2003*), pp. 225–233 (May 2003).

[13] B. Chidlovskii: Information Extraction from Tree Documents by Learning Subtree Delimiters, in *Proceedings of Workshop on Information Integration on the Web*(*WIIW 2003*), pp. 3–8 (Aug. 2003).

[14] W. W. Cohen, M. Hurst, and L. S. Jensen: A Flexible Learning System for Wrapping Tables and Lists in HTML Documents, in *Proceedings of International World Wide Web Conference* (*WWW 2002*), pp. 232–241 (May 2002).

[15] 日本語形態素解析器 ChaSen, *http://chasen-legacy.sourceforge.jp/*. (2007/9/20 アクセス)

[16] CNN, *http://www.cnn.com/*.

[17] M. Diligenti, M. Gori, and M. Maggini: A Unified Probabilistic Framework for Web Page Scoring System, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 16, No. 1, pp. 4–16 (Jan. 2004).

[18] D. R. Cutting, D. R. Karger, J. O. Pedersen, and J. W. Turkey: Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections, in *Proceedings*

of International Conference on Research and Development in Information Retrieval (*ACM SIGIR 1992*), pp. 318-329 (June 1992).

[19] DION ニュースサイト, *http://newstopics.dion.ne.jp/pubnews/*.

[20] H. Edmundson: New Methods in Automatic Abstracting, *Journal of the Association for Computing Machinery*, Vol. 16, No. 2, pp. 264-285 (Apr. 1969).

[21] 福田賢治, 石野 明, 竹田正幸, 松尾文碩: 生垣上の反単一化を用いた情報抽出手法の提案, 第 53 回知識ベースシステム研究会 (SIG-KBS) 論文集, pp. 47–52 (Sept. 2001).

[22] Google(日本), *http://www.google.co.jp/*.

[23] Google Wireless Transcoder, *http://www.google.com/xhtml/*.

[24] 服部 元, 松本一則, 菅谷史昭: 表形式情報集約のための人に優しい検索キーの自動抽出方式の検討, 合同エージェントワークショップ＆シンポジウム (JAWS 2004) 論文集, Vol. 2004, No. 32, pp. 425–432 (Oct. 2004).

[25] 服部 元, 松本一則, 菅谷史昭, 小池 淳, 山田 満, 坂村 健: 統合 PDA 端末の開発 (3)〜携帯端末のための Web ページ自動分割, 電子情報通信学会総合大会 通信講演論文集, Vol. 2005, No. 1, p. 714 (Mar. 2005).

[26] 服部 元, 松本一則, 菅谷史昭: タグの深さに基づくコンテンツ間距離を利用した Web ページの自動分割方式, 電子情報通信学会データ工学ワークショップ (DEWS 2005) 論文集, *http://www.ieice.org/˜de/DEWS/DEWS2005/procs/papers/6A-o3.pdf* (Mar. 2005).

[27] 服部 元, 松本一則, 菅谷史昭: タグの深さを利用したコンテンツ間距離に基づく Web ページの自動分割方式, 日本データベース学会 Letters, Vol. 4, No. 1, pp. 149–152 (June 2005).

[28] 服部 元, 松本一則, 菅谷史昭: 表形式情報集約のための連想性の高いオブジェクトラベルの自動抽出方式, 電子情報通信学会論文誌, Vol. J88-DI, No. 9, pp. 1467–1476 (Sept. 2005).

[29] 服部 元, 松本一則, 菅谷史昭: コンテンツ間距離の分布に基づく Web ページ動的分割方式, 情報処理学会データベースと Web 情報システムに関するシンポジウム (DBWeb 2005) 論文集, Vol. 2005, No. 16, pp. 73–80 (Nov. 2005).

[30] 服部 元, 松本一則, 菅谷史昭: コンテンツ間距離の標準偏差に基づく Web ページ動的分割方式, 情報処理学会論文誌：データベース, Vol. 47, No. SIG 8 (TOD 30), pp. 81–89 (June 2006).

[31] G. Hattori, K. Hoashi, K. Matsumoto, and F. Sugaya: Robust Web Page Segmentation for Mobile Terminal Using Content-Distances and Page Layout Information, in *Proceedings of International World Wide Web Conference* (*WWW 2007*), pp. 361–370 (May 2007).

[32] 服部 元, 原 隆浩, 菅谷史昭, 西尾章治郎: クリック型 Web 検索のための重要語推定方式, 情報処理学会データベースと Web 情報システムに関するシンポジウム (DBWeb 2007) 論文集 (CD-ROM), Vol. 2008, No. 1A-3 (Nov. 2007).

[33] 服部 元, 原 隆浩, 滝嶋康弘, 西尾章治郎: 周辺語を活用したクリック型 Web 検索システムの提案と評価, 情報処理学会論文誌：データベース, Vol. 1, No. 2, pp. 26–37 (Sept. 2008).

[34] スマートセンサー, *http://k-tai.hitachi.jp/w51h/smartsensor.html*.

[35] 石田栄美, 安形 輝, 久野高志, 上田修一: 情報源となりうる Web ページの判定, 日本図書館情報学会研究大会講演論文集, Vol. 2000, pp. 50–53 (Nov. 2000).

[36] 石谷康人, 鈴木 優, 布目光生: 連鎖検索と近傍検索に基づく Web コンテンツへの効率的なアクセス方法, 電子情報通信学会研究報告 (Web インテリジェンスとインタラクション研究会 WI2-2006-6), Vol. 2006, No. 6, pp. 31-36 (July 2006).

[37] J. F. Gantz, C. Chute, A. Manfrediz, S. Minton, D. Reinsel, W. Schlichting, and a. Toncheva, The Diverse and Exploding Digital Universe, IDC White Paper, *http://japan.emc.com/collateral/analyst-reports/diverse-exploding-digital-universe.pdf* (Mar. 2008).

[38] M. Jones, G. Buchanan, and H. Thimbleby: Sorting Out Searching on Small Screen Devices, in *Proceedings of International Conference on Human-Computer Interaction with Mobile Devices and Services* (*Mobile HCI 2002*), pp. 81–94 (Sept. 2002).

[39] The Japan Times Online, *http://www.japantimes.co.jp/*.

[40] 木本晴夫: 日本語新聞記事からのキーワード自動抽出と重要度評価, 電子情報通信学会論文誌, Vol. J74-D-I, No. 8, pp. 556-266 (Aug. 1991).

[41] Y. Kitamura, T. Noda, and S. Tatsumi: Single-Agent and Multi-Agent Approaches to WWW Information Integration, in *Proceedings of the Pacific Rim International Workshop on Multi-Agents* (*PRIMA 1998*), pp. 133–147 (Nov. 1998).

[42] C. A. Knoblock, S. Minton, J. L. Ambite, N. Ashish, I. Muslea, A. G. Philpot, and S. Tejada: The ARIADNE Approach to Web-based Information Integration, *International Journal of Cooperative Information Systems*, Vol. 10, No. 1/2, pp. 145–169 (Mar./June 2001).

[43] 小林拓海, 佐藤大介, 三末和男, 田中二郎: Web 検索結果の概観提示による情報収集支援インタフェース, 第 19 回人工知能学会全国大会 講演論文集, *http://www-kasm.nii.ac.jp/jsai2005/schedule/pdf/000145.pdf* (June 2005).

[44] KDDI 株式会社, *http://www.kddi.com/*.

[45] H. Lam, P. Baudisch: Summary Thumbnails: Readable Overviews for Small Screen Web Browsers. in *Proceedings of Conference on Human Factors in Computing Systems* (*CHI 2005*), pp. 681-690 (Apr. 2005).

[46] H. P. Luhn: A Statistical Approach to Mechanized Encoding and Searching of Literary Information, *IBM Journal of Research and Development*, Vol. 1, No. 4, pp. 390-317 (Oct. 1957).

[47] LinkShare 社, *http://www.linkshare.com/*.

[48] 前川卓也, 上向俊晃, 原 隆浩, 西尾章治郎: 複数のモバイル端末による協調ブラウジングのための木構造型コンテンツ記述方式と分割方式, 情報処理学会論文誌 : データベース, Vol. 45, No. SIG 7 (TOD 22), pp. 11–23 (June 2004).

[49] 前川卓也, 原 隆浩, 西尾章治郎: 複数のモバイルユーザのための Web ページ分割を用いた協調 Web ブラウジングシステム, 情報処理学会研究報告 (モバイルコンピューティングとユビキタス通信研究会 2004-MBL-31), Vol. 2004, No. 114, pp. 1–7 (Nov. 2004).

[50] 松尾 豊, 石塚 満: 語の共起の統計情報に基づく文書からのキーワード抽出アルゴリズム, 人工知能学会論文誌, Vol. 17, No. 3, pp.217-223 (Apr. 2002).

[51] 増田英孝, 塚本修一, 安富大輔, 中川裕志: HTML の表形式データの構造認識と携帯端末表示への応用, 情報処理学会論文誌：データベース, Vol. 44, No. SIG 12 (TOD 19), pp. 23–32 (Sept. 2003).

[52] 松下光範, 笹野直幸, 前田英作: 差分による WEB ページからの情報抽出のための基礎検討, 人工知能学会研究報告 (人工知能基礎論研究会 SIG-FAI, 知識ベースシステム研究会 SIG-KBS), pp.103–108 (Nov. 2001).

[53] 水口 充, 梅本あずさ, 柴尾忠秀, 浦野直樹: 提示型ユーザインタフェースの実装と評価, 日本ソフトウェア科学会学会誌：コンピュータソフトウェア, Vol. 18, No. 1, pp. 169–183 (Jan. 2001).

[54] 中藤哲也, 大森敬介, 廣川左千男: 検索エンジンを部品とするエージェントの構成について, 合同エージェントワークショップ&シンポジウム (JAWS 2004) 論文集, Vol. 2004, No. 32, pp. 441–445 (Oct. 2004).

[55] 大西健史, 荒瀬由紀, 原 隆浩, 上向俊晃, 西尾章治郎: 携帯電話ユーザのための加速度センサを用いた Web 閲覧システム, 電子情報通信学会データ工学ワークショップ (DEWS 2007) 論文集, *http://www.ieice.org/˜de/DEWS/DEWS2007/pdf/b9-1.pdf* (Feb./Mar. 2007).

[56] Opera Software ASA, *http://www.opera.com/products/mobile/smallscreen/*.

[57] OneStat.com, *http://www.rankstat.com/html/en/seo-news1-most-people-use-2-word-phrases-in-search-engines.html*.

[58] G. Salton, and M. J. McGill: Introduction to Modern Information Retrieval, McGraw-Hill College, 448 pp. (Sept. 1983).

[59] Target 社, *http://www.target.com/*.

[60] 社団法人電気通信事業者協会, *http://www.tca.or.jp/index.html*.

[61] 梅原雅之, 岩沼宏治, 永井宏和: 事例に基づく HTML 文書から XML 文書への半自動変換 – シリーズ型 HTML 文書における類似性の利用 –, 人工知能学会論文誌, Vol. 16, No. 5, pp. 408–416 (Feb. 2001).

[62] 若木裕美, 正田備也, 高須淳宏, 安達 淳: 検索語の曖昧性を解消するキーワードの提示手法, 日本データベース学会 Letters, Vol. 4, No. 2, pp. 37–40 (Sept. 2005).

[63] J. Wobbrock, J. Forlizzi, S. Hudson, and B. Myers: WebThumb: Interaction Techniques for Small-Screen Browsers, in *Proceedings of Symposium on User Interface Software and Technology* (*UIST 2002*), pp. 205–208 (Nov. 2002).

[64] J. M. Wu, T. Uemukai, and F. Sugaya: A User-Customizable Xwindow-like GUI System for Mobile Phones (Poster Paper), in *Proceedings of International Conference on Computer Graphics and Interactive Techniques* (*ACM SIGGRAPH 2006*) (CD-ROM), No. 145 (Aug. 2006).

[65] 山田誠二, 中井有紀: 対話的分類学習による Web ページの部分更新モニタリング, 人工知能学会論文誌, Vol. 17, No. 5, pp. 614–621 (July 2002).

[66] Yahoo! Japan カテゴリ, *http://www.yahoo.co.jp/*.

[67] Yahoo! Japan ウェブ検索 Web サービス, *http://developer.yahoo.co.jp/search/*.