Web アプリケーションの結果ページからの結果部分抽出手法

中野 雄介† 山登 庸次† 武本 充治† 須永 宏†

†日本電信電話株式会社 NTT ネットワークサービスシステム研究所 〒180-8585 東京都武蔵野市緑町 3-9-11

E-mail: †{nakano.yuusuke, yamato.yoji, takemoto.michiharu, sunaga.h}@lab.ntt.co.jp

あらまし 近年，Web2.0 のコンセプトの一部である，マッシュアップによる様々なコンポーネントが連携する，サービス提供が始まっている．このため，既存の Web アプリケーションをコンポーネント化するためのラッパ生成が必要となっている．そこで，Web アプリケーションが生成する HTML ドキュメントの中から，Web アプリケーションの処理結果の部分を抽出する手法を提案し，本手法のラッパ生成への応用を検討する．本手法は HTML ドキュメント内の各タグのネストの回数 (深度)の変化に規則的なパターンがある部分を結果部分として抽出する．本手法を評価するためのプロトタイプを実装し，既存の Web アプリケーションが生成する HTML ドキュメントから，結果部分の抽出を試みた．100 ドキュメント中 76 ドキュメントから結果部分を抽出することに成功した．

キーワード WWW，ラッパ，Web サービス，Web マイニング

Method of extraction search results from HTML documents generated by web applications

Yusuke NAKANO† Yoji YAMATO† Michiharu TAKEMOTO† and Hiroshi SINAGA†

†NTT Network Service Systems Laboratories, NTT Corporation

9-11, Midori-Cho 3-Chome Musashino-Shi, Tokyo 180-8585 Japan

E-mail: †{nakano.yuusuke, yamato.yoji, takemoto.michiharu, sunaga.h}@lab.ntt.co.jp

Abstract Services created by composing various components are provided these days. This service creation manner is called Mashup and this is one of Web 2.0 concepts. Thus, a wrapper that makes web applications usable as web services begins to be popular. In this paper, we propose a method of extraction search results from HTML documents generated by web applications and describe a possibility of the wrapper implemented by the proposed method. The proposed method extracts search results using characteristic depth pattern of each tag in the HTML documents. We implemented a prototype system to evaluate the proposed method and the evaluation results showed that the proposed method succeeded the extraction for 76 % of HTML documents generated web applications.

Keyword WWW, Wrapper, Web service, Web mining

1. はじめに

ユビキタスコンピューティング環境におけるユーザへのサービス提供に関する研究が盛んになってきており，利用可能なユビキタスサービスプラットフォームやそれらの上でのアプリケーションもある [1, 2, 3, 4]．このようなプラットフォームは Service Oriented Architecture (SOA) をベースとしている．つまり，サービスを構成するコンポーネントのインタフェースとして Web Service [5]を，インタフェースの記述には Web Service Description Language (WSDL) [6]を用い，各コンポーネントは Simple Object Access Protocol (SOAP) [7]により連携する．しかし，ユビキタスサービスユーザのコンテキストに対して最適なサービスを提供するためには膨大なコンポーネントが必要となる．

一方，マッシュアップと呼ばれる簡易なサービス作成手法が注目を集めている．マッシュアップは Web2.0 のコンセプトに含まれ [8]，近年のユーザによるサービス作成を促進している．マッシュアップによるサービス作成では，作成者は Web サービスなどのプログラムコンポーネントを発見し,それらをつなぎ合わせる．これにより,容易にサービスを作成することができる．

以上のように，Web 上で提供されるコンポーネントに注目が集まっている．そこで，我々は既存の Web アプリケーションをコンポーネントとして利用するためのラッパに関する研究を行ってきた．Web アプリケーションとは Web ブラウザの利用者から HTTP でリクエ

ストを受け，これに対する結果を HTML ドキュメントの形で Web ブラウザに返すことで働く Web 上のアプリケーションである．たとえば，ホテル検索や路線検索など，多くの Web アプリケーションが提供されている．我々のラッパはこのような Web アプリケーションをコンポーネント化するために，Web アプリケーションのプロトコルとコンポーネントのプロトコル(SOAP など)とを相互変換する．

多くの Web アプリケーションをコンポーネント化するためにラッパは Web アプリケーションごとに用意されたコンフィグファイルに沿って変換の処理を行う．例えば，ホテル検索 Web アプリケーションをコンポーネント化する場合は，ホテル検索用のコンフィグファイルを記述する必要がある．このため，多くの Web アプリケーションをコンポーネント化するためには膨大なコンフィグファイルを記述する必要があり，この作業の削減が求められていた．

コンフィグファイルの記述において，最も時間のかかる記述が，Web アプリケーションのプロトコルからコンポーネントのプロトコルへの変換ルールである．このルールは Web アプリケーションが生成する HTML ドキュメント内のコンポーネントの戻り値にあたる部分を指定する．つまり，Web アプリケーションが生成する HTML ドキュメント内の，Web アプリケーションの処理結果の部分を自動抽出できれば，コンフィグファイルの記述コストを大幅に削減できる．

そこで，本稿では Web アプリケーションが返す HTML ドキュメントから Web アプリケーションの処理結果の部分を自動抽出するための手法を提案し，提案手法の有効性を検証する．

2. 関連研究

様々な Web アプリケーションが異なる形式の HTML ドキュメントを生成する．このため，すべての形式の HTML ドキュメントから結果部分を抽出するラッパの実現は不可能である．しかし，膨大な Web アプリケーションそれぞれにラッパを作成するためには多大なコストがかかる．そこで近年，Web ラッパの生成に関する研究が盛んになっている．

[9]で Web ラッパの生成に関する研究がまとめられている．Web ラッパの生成は教師つき学習と教師なし学習との 2 つに分類でき，教師つき学習である[10]からラッパ生成に関する研究が盛んになった．本稿で提案する手法は教師なしの学習に分類されるため，教師無し学習によるラッパ生成に関する関連研究をあげる．

[11] は Web アプリケーションのソースコードからラッパを生成する．成功率は高いが，ソースコードが必要となるため，市中にある多くの Web アプリケーションの Web サービス化はできない．

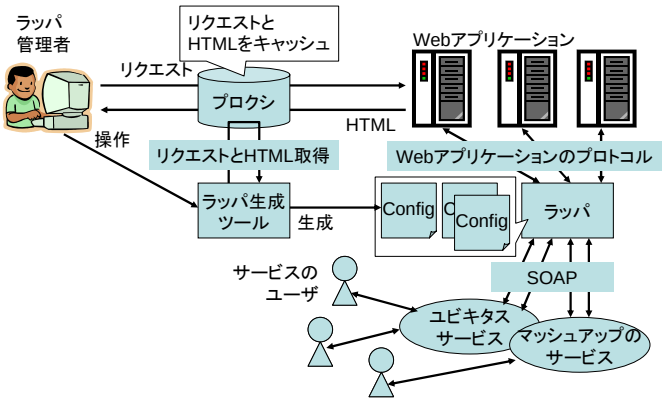

図 1　ラッパシステム概要

[12] は文字列の最大反復を発見することにより，結果部分を抽出するラッパを提案している．この手法は結果部分にイレギュラーな結果 (宿検索の結果ページ内の，お勧めの宿の結果だけタグの構造が違うなど)が含まれた場合，抽出に失敗すると考えられる．

また，近年実際に Web アプリケーションをコンポーネント化するためのラッパ生成ツールの提供が始まっており [13]，このようなラッパに対して注目が集まっている．

3. ラッパシステム

我々のラッパは多くの Web アプリケーションをラップするために，コンフィグファイルを必要とする．コンフィグファイルを手で記述することができるが，多大な労力を必要とする．この課題を解決するためのラッパシステムについて述べる．

3.1. ラッパシステムの概要

ラッパシステムの概要を図 1 に示す．まず，ラッパ管理者はラップ対象の Web アプリケーションを専用のプロクシサーバを介して利用する．このとき，プロクシサーバにはラッパ管理者の Web ブラウザから Web アプリケーションへのリクエストとリクエストに対する HTML ドキュメントが保存される．その後，ラッパ管理者はラッパ生成ツールを操作することで，プロクシサーバ内のラップ対象の Web アプリケーションへのリクエストと HTML ドキュメントからコンフィグファイルを生成する．生成されたコンフィグファイルによってラッパはラップ対象の Web アプリケーションをラップできる．このようにして，Web アプリケーションをコンポーネント化でき，様々なユビキタスサービスやマッシュアップサービスで利用可能となる．

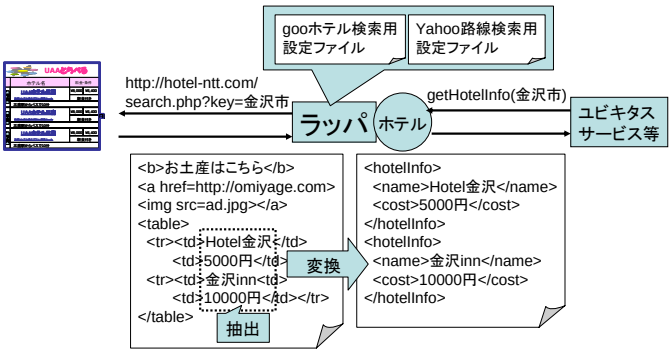

図 2 ラッパ動作

3.2. ラッパ

ラッパは Web アプリケーションのプロトコルと，コンポーネント(ここでは Web サービスとする)のプロトコル(ここでは SOAP とする)とを相互変換することで Web アプリケーションをコンポーネント化する．図 2 にラッパの動作を示す．まず，ラッパはユビキタスサービスなどの Web サービスクライアントから SOAP でリクエストを取得する．その後，ラッパは SOAP で取得したリクエストを Web アプリケーションのリクエストに変換し，変換後のリクエストを Web アプリケーションへ送信する．リクエストを受け取った Web アプリケーションはそれに対するレスポンスを HTML ドキュメントの形でラッパへ送信する．HTML ドキュメントを受け取ったラッパは HTML ドキュメントからか Web アプリケーションの処理結果の部分を抽出し，これを XML ドキュメントの形に整形する．その後，この XML ドキュメントを Web サービスの戻り値としてクライアントに返す．このようにして，ラッパは Web アプリケーションを Web サービス化する．以上の処理はラッパに設定されたそれぞれの Web アプリケーション用のコンフィグファイルに沿って行われる．

4. 提案手法

コンフィグファイルの生成のために Web アプリケーションが生成する HTML ドキュメントから，Web アプリケーションの処理の結果の部分のみを抽出する手法を提案する．提案手法により，Web アプリケーションのプロトコルからコンポーネントのプロトコルへの変換ルールの生成を支援できる．

4.1. 提案手法の概要

提案手法は Web アプリケーションが生成する HTML ドキュメントに含まれるタグの深度変化の特徴を手がかりとし，結果部分を抽出する．タグの深度とはタグの入れ子の回数である (図 3(1))．Web アプリケーションは機械的に HTML ドキュメントを生成する．このため，生成された HTML ドキュメントには繰り返しのパターンが現れることが多い (図 3(2))．このような Web アプリケーションが生成する HTML ドキュメントの特徴を利用する．

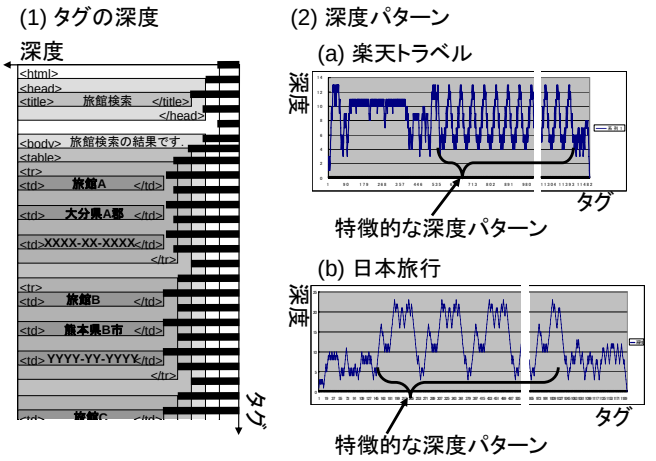

図 3 タグ深度とパターン

本提案手法は人が Web アプリケーションの結果ページから結果部分を見つける方法と類似している．ある人が海外のホテル検索サイトの結果ページからホテルの検索結果を探し出すとする．この人は結果のページから繰り返し同じようなパターンが連続している範囲を探す．この方法はうまく機能し，この人はホテルの検索結果を見つけることができる．このような，人が自然に身に付けている方法を用いることで，様々な Web アプリケーションに適用可能な，柔軟な抽出を実現できる．

4.2. 提案手法の詳細

提案手法は HTML ドキュメントに含まれるタグの深度が周期的に変化する部分を結果部分であると推定する．これを実現するために，深度データを波と考え，この波をスペクトル分析することで，周期的に変化している部分を発見する．発見された部分は波全体における大まかな位置を示すため，推定された位置と HTML 内のタグの情報とを用いて，HTML 上の正確な結果部分の位置を推定する．

HTML ドキュメントから結果部分を抽出するための手法を図 4 に示す．本手法は 6 つのステップから構成される．各ステップについて説明する．

Step1. HTML ドキュメント内の各タグの入れ子の回数をカウントすることで深度を算出し，深度データを生成する．

Step2. HTML ドキュメント内の各部の深度特徴を抽出するために，算出された深度データを当分する．

Step3. FFT により，等分された深度データを周波数成分に展開し，周波数特性を各部の深度特徴とする．

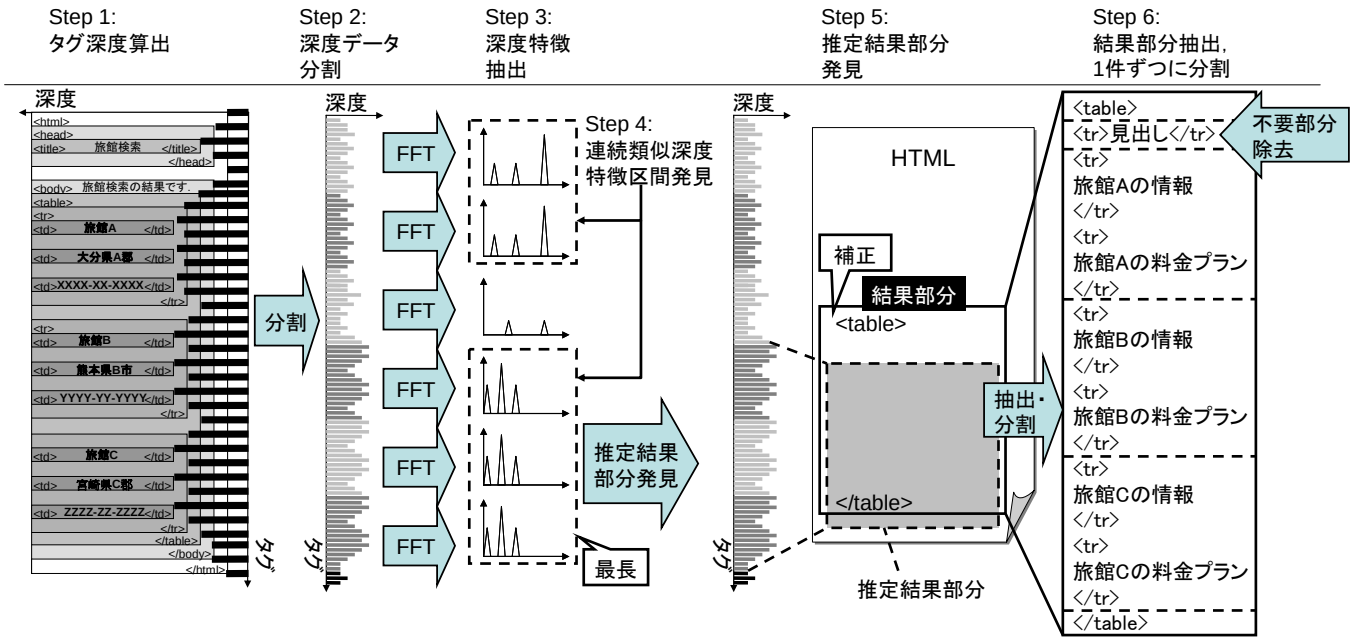

図 4　結果部分抽出手法

Step4. HTML 内で，深度特徴が類似する部分が連続している区間 (連続類似深度特徴区間) を発見する．

Step5. 発見された連続類似深度特徴区間で長さが最長となる区間に対応する HTML ドキュメント内の部分を推定結果部分として発見する．

Step6. 推定結果部分を補正し，正しい結果部分として抽出し，結果部分に含まれる結果を検索結果 1 件ごとに分割する．このとき，表の見出しなど，結果以外の不要部分を除去する．

次からは各ステップに関して述べる．

4.2.1. タグ深度算出

Web アプリケーションが生成する HTML ドキュメント内の各タグの入れ子回数をカウントする．このとき，カウントされるタグは開始タグと終了タグのペアからなるタグである．つまり，
や<img>など，終了タグを伴わないタグを無視する．これは，終了タグを伴わないタグは不規則に出現する傾向があるからである．例えば，宿検索サイトの結果ページに，各宿に関するコメントが記述されていたとする．このようなコメントには，
タグが多数含まれ，且つ，それぞれの宿のコメントによって，含まれる
タグの数は異なると予想できる．このような，不規則に用いられるタグは，周期的な深度パターンを崩す可能性がある．

4.2.2. 深度データ分割

HTML ドキュメントの各部の深度特徴を算出するために，深度データを等分する．これにより，部分ごとの深度特徴を調べることができ，波全体から周期的に深度が変化している部分を発見することができる．

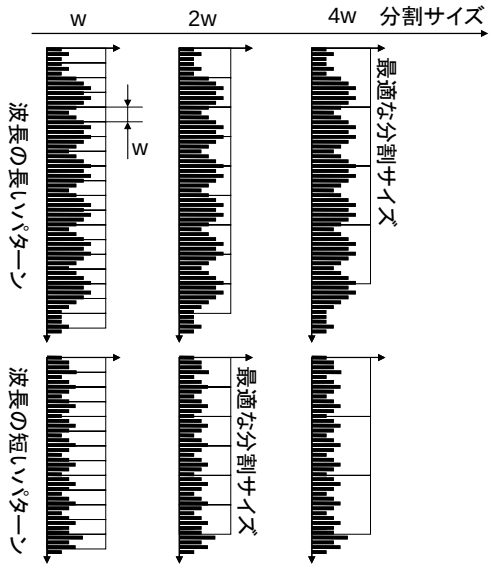

図 5　深度データ分割サイズ

分割のサイズは HTML ドキュメントの深度特徴により，可変とする必要がある．これは，図 4 に示すように，HTML ドキュメントに含まれるパターンが，波長の長いパターンを持つ場合は分割サイズを広くし，波長の短いパターンを持つ場合は分割サイズを狭くする必要があるためである．

分割サイズは基準となるサイズを決めておき，まずそのサイズで分割を行い，結果部分の抽出を試みる．その後，基準サイズの 2 倍のサイズで分割し，再度抽出を試みる．このようにして，基準のサイズの 2 乗倍

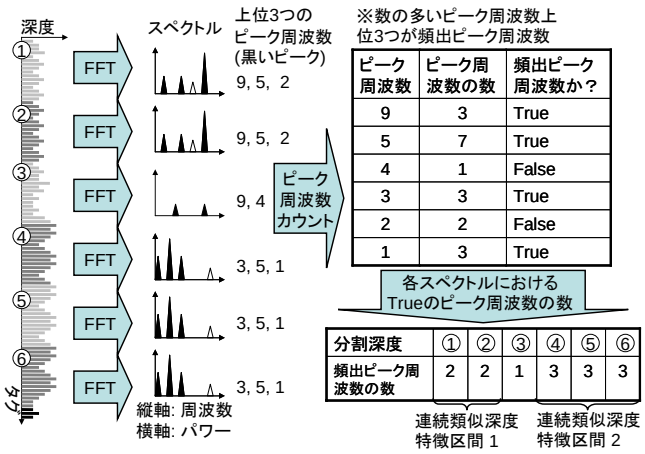

図 6　結果部分の推定

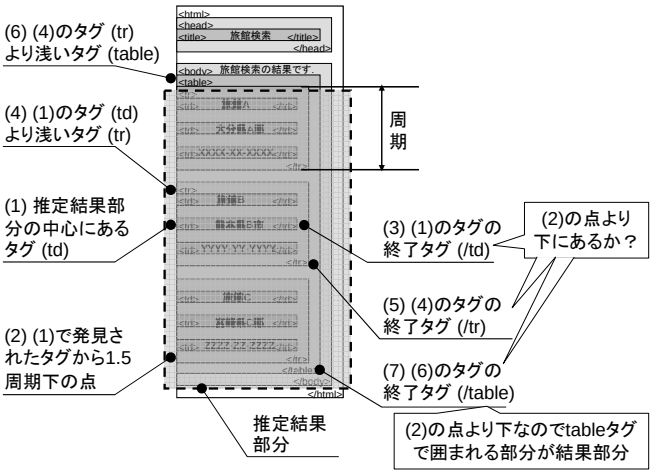

図 7　結果部分の抽出

のサイズで分割を行い，抽出を試み，さらに 2 倍のサイズで分割し，抽出するという動作を繰り返す．分割サイズが波全体の 4 分の 1 になると，サイズの拡大を停止し，それまでに抽出された結果部分から最長のものを正しい結果部分と判断する．

このようにして，様々な Web アプリケーションに最適なサイズの分割を行うことで，適切に結果部分を抽出することができる．

4.2.3. 深度特徴抽出

FFT により，分割された深度データそれぞれを周波数成分に展開し，各分割深度の深度特徴を抽出する．深度特徴は，FFT によって算出されたスペクトルをから，高いピーク周波数の上位 3 つとする．図 6 を用いて説明すると，分割深度①のスペクトルにはピークが 4 つあり，そのうち上位 3 つ (黒く塗りつぶされたピーク) の周波数は 9，5，2 であった．つまり，部分深度①の深度特徴は［9，5，2］となる．

4.2.4. 連続類似深度特徴区間発見

深度特徴が類似している分割深度が連続している部分を発見する．この部分を連続類似深度特徴区間と呼ぶ．連続類似深度特徴区間の発見手法を図 6 で説明する．まず，全分割深度の深度特徴に含まれるピーク周波数を周波数ごとにカウントする (周波数 6 は④，⑤，⑥の深度特徴に含まれるので，3 回とカウントされる)．次に，それぞれのピーク周波数が頻出するかどうかを確認する．本手法では，カウント回数の多い上位 3 つのピーク周波数を頻出ピーク周波数とする (1 位は 7 回の周波数 5．2 位・3 位は 3 回の周波数 9，3，1)．その後，各分割深度に含まれる頻出ピーク周波数の数をカウントする (分割深度①の場合，ピーク周波数 9，5 が頻出ピーク周波数であるため，2 とカウントされる)．各分割深度の深度特徴に含まれるピーク周波数の過半数が頻出ピーク周波数となる分割深度を連続類似深度特徴区間とする (分割深度①，②は頻出ピーク周波数を 2 つ含むため，連続類似深度特徴区間となる)．

4.2.5. 推定結果部分発見

複数の連続類似深度特徴区間が発見された場合，最長の区間を推定結果部分とする．これは，結果部分は Web アプリケーションが生成する HTML ドキュメントの広い部分を占めると仮定したためである．このようにして，発見された推定結果部分は HTML ドキュメント中での大まかな結果の位置を示す (図 6 の場合，分割深度④～⑥に当たる部分が推定結果部分となる)．

推定結果部分の発見と同時に，結果部分の波の周期を算出する．これは推定結果部分の深度データのスペクトル中，最も高いピーク周波数の逆数である．この周期は結果部分に含まれる結果ひとつに含まれるタグの数と等しい (図 7 の周期)．

4.2.6. 結果部分抽出

推定結果部分は結果部分の大まかな位置を示すため，正確な結果部分を抽出する必要がある．これには，推定結果部分内にあるタグの情報を用いる．これにより，推定結果部分を補正し，結果部分を抽出することができる．

図 7 に結果部分抽出のための手法を示す．はじめに，推定結果部分の中心にあるタグを発見する (図 7 では td)．次に，発見されたタグから 1.5 周期下の点を発見する．その後，発見されたタグの終了タグを発見する (図 7 では/td)．このとき，発見された終了タグが，開始タグから 1.5 周期下の点より上にあるとき，開始タグより浅いタグを発見し，新たな開始タグとする．そして，新たな開始タグに対する終了タグを新たな終了

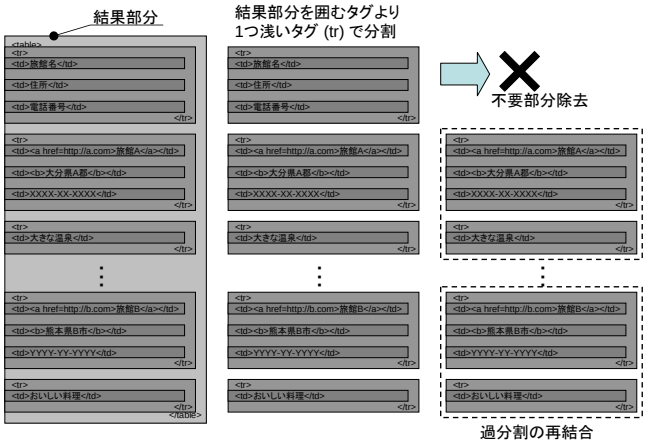

図 8　結果部分の分割

タグとし，新たな終了タグが先の点より下にあるかどうかを再度確認する．この処理を，終了タグが先の点より下になるまで繰り返す．終了タグが先の点より下になった場合，その終了タグ (図 7 では/table) とこれと対になる開始タグ (図 7 では table) とに囲まれる区間を結果部分とする．

4.2.7. 結果部分分割

抽出された結果部分を結果 1 件ごとに分割する．例えば，宿検索の Web アプリケーションの場合，宿 1 軒分の情報ごとに分割する必要がある．これにより，結果 1 件ごとに抽出できる抽出ルールを生成可能となる．

分割は結果部分を囲むタグ (図 8 の場合 table タグ) よりも 1 つ浅いタグ (図 8 の場合 tr タグ) がある箇所で行われる．しかし，このような分割では，タイトルのような結果以外の不要部分が混入していたり，1 件分の結果が複数に過分割されたりするため (図 8 では宿の基本的な情報と，宿に対するコメントとで分割されている)，これらの課題の解決が必要となる．

4.2.8. 過分割の再結合

先のような課題を解決するために，過分割の再結合を行う．過分割の解決には，再結合数を算出する (図 8 の場合,1 回再結合することで旅館の情報とコメントとを結合でき，宿 1 軒ごとの結果に分割できる)．

図 9 に再結合数の算出手法を示す．まず，各分割結果をグラフと考える．次に，各グラフ同士でルートからリーフまでの一致回数をカウントする (A, B 間では tr-td-text が一致しているので 1 回)．その後，各グラフで一致回数があらかじめ定められた閾値以上になっている回数をカウントする (図 9 の類似数)．その後，類似数を縦軸，分割結果グラフを横軸とし，各分割結果グラフの類似数をプロットする．これを用い，プロットされた点から一次関数的に減少する点列を発見する．

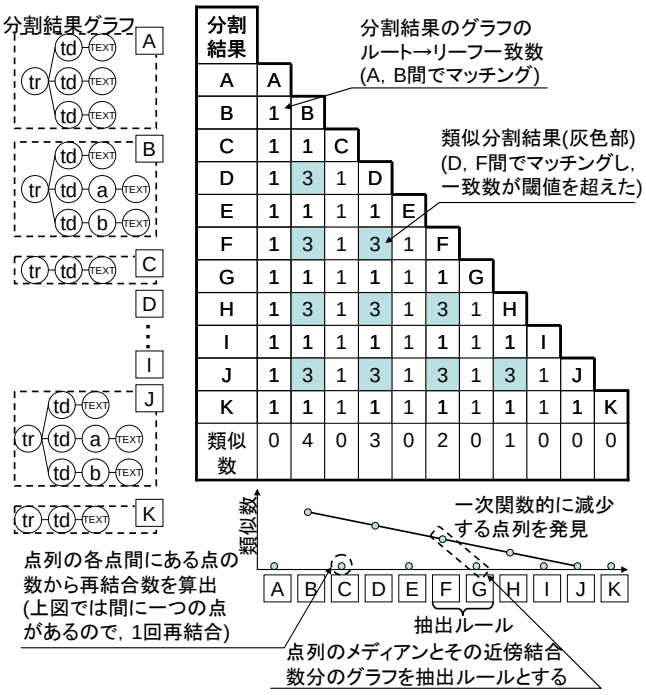

図 9　過分割の再結合

このとき，発見された点列の各点の間にある点の数を結合数とする (図 9 では各点の間に 1 つの点があるので，再結合数 1)．

4.2.9. 不要部分の除去

不要部分が混入する課題を解決するために，抽出ルールで抽出できない分割結果を不要部分と判定する．抽出ルールは抽出すべき結果であると推定される一つのグラフとする．このグラフと類似する分割結果を抽出すべき結果であると判定し，それ以外を不要部分と判定する．

抽出ルールは先で発見された点列のメディアンとその近傍の結合数分のグラフとなる (図 9 では F，G のグラフが抽出ルールとなる)．これは，メディアンが抽出すべき結果である可能性が高いと仮定したためである．また，結合数分の抽出ルールが必要となるので，メディアンに近い，つまり，抽出すべき結果である可能性の高いグラフも抽出ルールとする．このようにして生成された抽出ルールを結果部分全体に適用することで，不要部分を除去することができる．

5. 評価

我々は提案手法の有効性を確認するために，プロトタイプシステムとして，評価用のプログラムを実装し，既存の Web アプリケーションが生成する HTML ドキュメントから結果部分を抽出することによる提案手法の評価を行った．

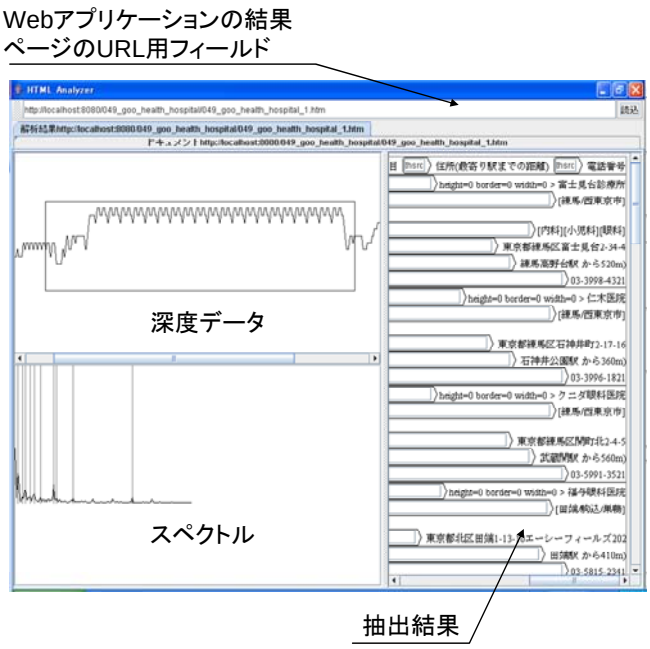

図 10　評価用プログラム

5.1. 評価手法

評価用プログラムのユーザインタフェースを図 10 に示す．評価者は Web アプリケーションが生成する HTML ドキュメントの URL を URL 入力用のフィールドに入力し，送信ボタンを押下する．すると，本プログラムが抽出対象となる HTML ドキュメントを取得し，先に説明した手法によって結果部分の抽出を試みる．その後，深度データ，スペクトルデータの可視化結果と，抽出結果とを表示する．

評価用の HTML ドキュメントは既存の Web アプリケーション 100 個が生成するドキュメントとした．対象とする Web アプリケーションは旅館検索等の検索サービスを提供するものであり，商品の購入などのサービスを提供する Web アプリケーションは対象外とした．また，本手法は HTML ドキュメントの規則的な深度パターンを手がかりとするため，検索結果が 5 件以上となるドキュメントを収集した．表 1 に収集されたドキュメントの Web アプリケーションの例を示す．

評価においては，正しい抽出ルールが生成できた場合に関して抽出成功とした．つまり，抽出結果に Web アプリケーションの処理の結果が，1 件ごとに結合された状態で，含まれている場合，この抽出結果を成功とした．

5.2. 評価結果

以上の評価手法による評価結果を図 11 に示す．64% の HTML ドキュメントに対して，過不足なく結果部分の抽出に成功している．8%のドキュメントに対しては抽出すべき一部の結果を不要部分と誤認しており，抽出に失敗している．さらに，4%のドキュメントに対しては不要部分が混入した．しかし，これら 12%に関しては抽出ルールを正しく生成できていると判断し，合計 76%のドキュメントに対して有効であると確認した．

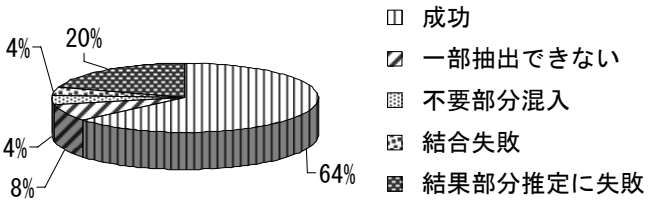

図 11　評価結果

結合に失敗している 4%の原因は，結合の順番を誤って判定してしまったことであった．これは，旅館情報の下に旅館のコメントを結合するべきだが，旅館のコメントの下に旅館の情報を結合してしまった，などを例としてあげることができる．これは，正しい抽出ルールとは言えない．

また，結果部分の推定に失敗している 20%に関しては，HMTL ドキュメント中に複数の周期的な深度パターンが現れることが主な原因であった．例えば，結果ページに大量のリンク集が合った場合，この部分に周期的な深度パターンが現れる．提案手法では，リンク集と Web アプリケーションの処理結果の部分を見分けることはできないため，誤ってリンク集の部分を抽出してしまうことがある．

6. おわりに

本稿では，Web アプリケーションの結果ページから，結果部分を抽出するための手法に関して述べた．本手法は Web アプリケーションをコンポーネントとして利用可能とする，ラッパを動作させるコンフィグファ

表 1　評価対象 Web アプリケーション

Web アプリ	URL	抽出成否
楽天市場	http://www.rakuten.co.jp/	OK
楽天トラベル	http://travel.rakuten.co.jp/kaigai/	OK
Yahoo ニュース	http://headlines.yahoo.co.jp/hl	推定結果部分発見失敗
Yahoo 中古車	http://autos.yahoo.co.jp/ucar/search/joken.html	OK
Yahoo アパート検索	http://realestate.yahoo.co.jp/	OK
goo 車・バイクカタログ	http://autos.goo.ne.jp/catalog/index.html	OK
goo 転職	http://job.goo.ne.jp/	OK
goo アニメ	http://anime.goo.ne.jp/tvanime/index.html	OK
MSN travel blog	http://4travel.travel.msn.co.jp/e/msn/travelogue/overseas/	推定結果部分発見失敗
Infoseek ニュース	http://news.www.infoseek.co.jp/	OK

イルの生成支援を実現する．本手法の評価の結果，既存の Web アプリケーションの 76%から正しいコンフィグファイルを生成できることを確認した．

今後は，本手法でコンフィグファイルの生成ツールを実装する．また，生成されたコンフィグファイルを用いたラッパを実現し，Web アプリケーションによって実現されたコンポーネントの組み合わせによる，サービスの実現を目指す．

謝辞

本研究の一部は，平成 17 年度総務省「ユビキタスネットワーク認証・エージェント技術の研究開発」の研究助成によるものである．

文　献

[1] M. Takemoto, H. Sunaga, K. Tanaka, H. Matsumura, and E. Shinohara, “The Ubiquitous Service-Oriented Network （USON） An Approach for a Ubiquitous World Based on P2P Technology”, in Proc. of P2P2002, pp.17-21, Sep. 2002

[2] M. Takemoto et al., “Service Elements and Service Templates for Adaptive Service Composition in a Ubiquitous Computing Environment”, in Proc. of Asia Pacific Conference on Communications (APCC), vol. 1, pp. 335-338, Sept. 2003.

[3] M. Takemoto, et al., “A Service-Composition and Service-Emergence Framework for Ubiquitous Computing Environments”, SAINT2004 Workshop, Jan. 2004.

[4] M. Takemoto et al., ”A Context-Aware Content-Provision Service Based on a Ubiquitous Service-Oriented Network Framework,” in Prof. of SAINT Workshop, Feb. 2005

[5] Web services web site, http://www.webservices.org/

[6] WSDL: W3C Note, “Web Services Description Language (WSDL) 1.1,” Mar. 2001, http://www.w3.org/TR/wsdl/

[7] SOAP: W3C Note “Simple Object Access Protocol (SOAP)”, 1.1 May 2000 http://www.w3.org/TR/soap/

[8] Tim O'Reilly, “What Is Web 2.0”, http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html, Oct. 2005.

[9] 山田泰寛，池田大輔，坂本比呂志，有村博紀，“WWW からの情報抽出－Web ラッパーの自動構築－”，人工知能学会誌，19 巻，3 号，pp. 302-309，2004 年．

[10][10] N. Kushumerick, “Wrapper Induction: Efficiency and Expressiveness”, Artificial Intelligence, Vol. 118, No.1-2, pp. 15-68, 2000.

[11]H. P. Huy, "Web Service Gateway − a step forward to e-business", in Proc. ICWS '04, pp. 648-655, June 2000

[12]C. H. Chang, and S. C. Lui, “IEPAD: Information Extraction Based on Pattern Discovery”, Proc. 10th International Conference of World Wide Web, pp. 4-15, 2001.

[13]Dapper, http://www.dappit.com/index.php