Keio WIX システム (3) コンテンツ作成

市東　隼† 分部 亮太† 朱　成敏† 遠山元道††

† 慶應義塾大学理工学部情報工学科 〒 223–8522 神奈川県横浜市港北区日吉 3–14–1
E-mail: †{sitow,wake,joo}@db.ics.keio.ac.jp, ††toyama@ics.keio.ac.jp

あらまし Web Index(WIX) ファイルとはキーワードとそれに対応する URL からなるエントリを持つ XML ファイルである．それを閲覧中の Web 文書に結合（アタッチ）すると，文書中のキーワード部分が対応する URL へのハイパーリンクに変換され，ユーザはキーワードに関する新たな情報をリンク先の Web 文書から手にすることが出来る．WIX を利用に必要な WIX ファイルの作成方法には，手動で記述する方法と，指定した Web ページのリンク集から自動生成する方法が挙げられる．この２通りの作業を支援するシステムについて報告する．

キーワード Web Index，Web， 情報推薦

Keio WIX system (3) contents making

Hayato SITOW†, Ryota WAKEBE†, Sungmin JOO†, and Motomichi TOYAMA††

† ††Department of Information and Computer Science,
Keio University
Hiyoshi 3–14–1, Kouhoku-ku, Yokohama-shi, Kanagawa, 223–8522 Japan
E-mail: †{sitow,wake,joo}@db.ics.keio.ac.jp, ††toyama@ics.keio.ac.jp

1. はじめに

Web Index(WIX) とは，Web における結合可能な情報資源である．関係データモデルの表の「正規化」により，1 つの表を複数の表に無損失に分解を行うことで，情報量を損なわずにデータ量の削減と，データの整合性の維持が容易になる．こうした関係データモデルの特徴を Web に応用した WebIndeX(WIX) という研究を行っている．

現在の Web では，特定のアンカーテキストから特定のページへのリンクがされるという構造が一般的である．WIX では，アンカーテキストとリンクを Web ページから独立した「キーワードとリンク先の集合」(WIX ファイル) として扱い，任意のドキュメントに対してユーザ主導で「結合」することでドキュメント内の所々の文字列の詳細情報が記述された URL のハイパーリンクに変換する．WIX ファイルには以下のような利用方法がある．

辞書的な利用: Web ページを閲覧する際に知らない単語などを調べる目的での利用
誘導での利用: サイト作成者が自らのサイトへのリンクを記した WIX ファイルを作成，公開することにより自サイトへの誘導を図る
サイト作成ツールとしての利用: Web サイトを作成する際，WIX ファイルのアタッチでドキュメントのハイパーリンクを生成することによって，サイト作成の際のコストが軽減される

2. 研究目的

WIX ファイルの生成法には，手動による記述，Web ページのリンク集からの自動生成などが挙げられる．手動での記述の場合，ユーザーの意図が最も反映された WIX ファイルが作成できるが，手間と時間がかかる．用途に応じた WIX ファイルを作るにはリンク集からの自動生成が適しているが，リンク集があるページを探しさなければならず，抽出すべきでないリンク (ノイズ) を同時に抽出する可能性もある．また，新製品の追加・選手の移籍などのような情報の更新を WIX に即座に反映することができない．そこで，手動記述を補助し，手間を削減する支援システムの提案と，リンク集を自動的に発見して WIX ファイルを随時生成し，その情報元を監視するの２つシステムが必要と考え，WIX ファイルを効率的に生成・更新することを目的とする．

3. WIX プロジェクト

3.1 WIX の書式

WIX とは，Web における結合可能な情報資源である [2]. WIX ファイルは XML 形式で記述されたエントリの集合であり，記述例は図 1 のようになる. エントリ内には，キーワードとなる見出し語を keyword 要素に格納し，それに対応する詳細情報を示す参照先の URL を target 要素に格納する.

～野球選手 WIX の例～

```
<wix>
  <entry>
    <keyword>イチロー</keyword>
    <target>http://espn.com/ichiro.html</target>
  </entry>
  <entry>
    <keyword>松井</keyword>
    <target>http://espn.com/matsui.html</target>
  </entry>
  …
</wix>
```

図 1　WIX ファイルの記述例

3.2 アタッチ

実際にブラウザ上に表示された HTML 文書と WIX を結合する操作をアタッチと呼ぶ. 現在閲覧している文書に，WIX 内に記述されている見出し語が存在した場合，その文字列を見出し語に対応する URL 先へのハイパーリンクに変換する.

3.3 WIX のターゲットの性質

あるキーワードに対してターゲットとして WEB ページになり得るものは多数存在している. 例えばキーワードがスポーツ選手や芸能人などの氏名の場合, ブログや公式ページ, 写真, Wikipedia などが候補として挙げられる. 他にも難読語などのターゲットとして, 辞書ページなどが挙げられる. スポーツ選手のブログの中でも本人が書いたものや他人が書いたそのキーワードを含むものなど数多く存在する.

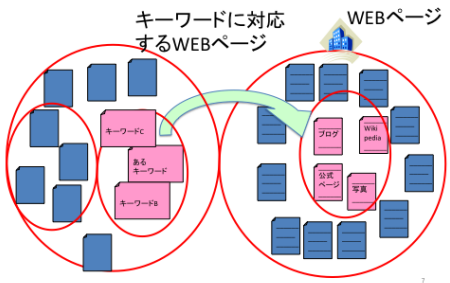

図 2　キーワードに対する WEB ページ候補の例

ターゲットは WIX ファイル作成者が作成しようとしている WIX ファイルに合わせて選び, 決定する. キーワードとターゲットの組み合わせをエントリとして WIX ファイルに記する. キーワードは共通点があるものをまとめて WIX ファイルとする. 例えば, 東野峻や阿部慎之助は野球選手や読売巨人軍という共通点がある. ターゲットも共通点があるものを選ぶ. 特に共通の性質を持つものを選ぶ. 例えば, 野球選手 WIX の時「ブログである」,「本人が書いている」などの共通点が部分集合になるものを選ぶ. 同一の WIX ファイルで対応付けるページの共通点が少ない場合, 利用者は選ぶキーワードごとに別のカテゴリのページに対応付けてしまう. そのため, 同一ファイルの WIX ファイルでは共通点が多いターゲットを選びカテゴリを統一することが求められる. 例えば, 野球選手 WIX の時, ある選手のターゲットのカテゴリがブログ, ある選手のターゲットのカテゴリが公式ページと選手によって違った場合, ターゲットカテゴリの統一の取れていない WIX ファイルと言える.

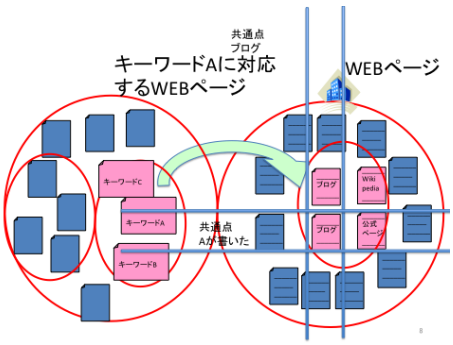

図 3　共通点のある WEB サイト

4. 手動記述支援システム

4.1 手動記述支援システムの情報提示手法

ここでは，本論文で提案する WIX ファイルの手動記述支援手法について述べる.WIX ファイル作成作業に対し，ターゲットとして記載する URL の発見や，新たに加えるべきエントリのキーワードの発見を補助するために情報提示を行う.

WIX ファイルの作成時の支援として以下のようなものが考えられる.

- キーワードのひらがな情報の提示
- キーワードに対する時期語の提示
- キーワードの WEB 検索結果の URL の提示
- 提示された URL と既に WIX ファイルに記されているターゲットとの類似度による推薦
- WIX ファイルのターゲットに成りやすい語をスニペットに含む URL の推薦

実際にはさらに無数の支援が考えられるが, 今回はこれらについての提案をおこなう.

4.1.1 ひらがな情報の提示

Yahoo デベロッパーネットワークのテキスト解析を使い入力されたキーワードに対応するひらがなの情報を検索する. ひらがな情報を WIX ファイルのキーワードに付け加えることで WEB ページ上でアタッチされる語が増え, キーワード緩和がおこなわれる. 難しい漢字や特殊な読み方の漢字などのキーワード緩和をおこなう時に有用である.

4.1.2 キーワードに対する時期語の提示

キーワードの時期語を得る web サービスとして kizasi.jp が挙げられる. この web サービスは毎月 25 万件以上ものブログ記事を時系列毎に要約し, ユーザが与えた一つのキーワードによってブログ記事を検索することでそのキーワードとともに用いられている関連語や, そのキーワードとともに語られている記事の話題を成分としたベクトルをユーザに返すというサービスである. ブログ記事を時系列毎に要約しているため, ホットなキーワードについては非常に多くの関連語が得られる. 例えば年末の時期に「甲子園」というキーワードを入力すれば「阪神」や「高校野球」といった関連語が出力されるし, 話題の成分として「感動」と「ワクワクする」といった成分が提示される. しかし, あまりホットではないキーワード, 例えば数年前に流行ったおもちゃの名前だとか事件の名前等を入力してもほとんど関連語は出てこない. また全てのブログにおいて一切語られていないキーワードは出力結果さえ返ってこない.kizasi.jp は本来, ある時期におけるホットな話題や関心を抽出するのが目的であるからあらゆるキーワードに対応しているわけではないということになる.

4.1.3 キーワードの WEB 検索結果の提示

Yahoo デベロッパーネットワークのウェブ検索機能を使い, タイトル, URL, スニペットを取得する. 手動での記述方式を支援する.

4.1.4 提示された URL と既に WIX ファイルに記されているターゲットとの類似度による推薦

すでに WIX ファイルに記されているターゲットの情報と WEB 検索結果で得られた URL を比較する. その結果, 関連度が高い URL を上位で推薦する. WIX ファイルのドメイン部とパス部に注目する. ドメイン部とは http://以降から次のスラッシュ(/) までの文字列と定義する. パス部とはドメイン部のスラッシュの後から次のスラッシュまでの文字列と定義する.
入力されたキーワードに対する WEB 検索をおこないタイトル, URL, スニペットを取得する. 得られた URL のドメイン部とパス部に記されている文字列と, 既に WIX ファイルに記されているドメイン部とパス部の比較をおこない一致している文字数を得る.WIX ファイル内の全てのエントリに対しておこない, パス部, ドメイン部の共通文字数が多い順にソートして提示する.WIX ファイルに求められるターゲットのカテゴリ一致を達成するためにドメイン部やパス部に注目する. ドメイン部とパス部が一致しているときは高確立で WIX ファイルのターゲットとなる.

4.1.5 WIX ファイルのターゲットに成りやすい語をスニペットに含む URL の推薦

WIX ファイルのターゲットになりやすい URL を推薦するために, WEB 検索をおこなう時に得られたスニペットを使う. スニペットとはキーワード検索して得られた summary の部分である. スニペットにターゲットの候補になりやすい語が含まれる場合, その URL を候補として推薦する. その時, 精度を高めるためにスニペットを形態素解析をおこなう.
ユーザーにターゲットの候補として成りやすい語を入力させる. 例えば, 「ブログ」, 「公式」など. 成りやすい語を入力させることによりユーザーの今作りたい WIX ファイルのターゲットカテゴリを推薦できる. 得られたスニペットを形態素解析した結果、もし入力されたターゲットに成りやすい語と一致する時その結果を上位にソートする.

4.2 提案した手動記述システムの外部仕様

（1） インプットとしてキーワードを指定.
（2） ターゲットになりやすい URL を推薦するために候補語を指定.
（3） 入力されたキーワードに対応するのひらがな情報を提示.
（4） 入力されたキーワードに対応する時期語情報を提示.
（5） 入力されたキーワードを Web 検索エンジンでの結果を提示.
（6） 得られた URL と既に WIX ファイルに記されているターゲットの類似度が高いものに目印を付け、上位にソートして提示. 比較した結果は図 5 に示すような形式で提示.
（7） ターゲットに成りやすい語を含むものに目印をつけて候補として提示. 比較した結果は図 6 に示すような形式で提示.
システムの外見は図 4 のようになる.

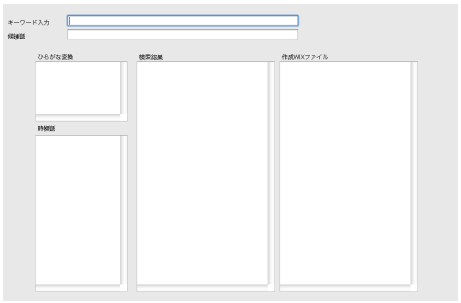

図 4 システムの外見

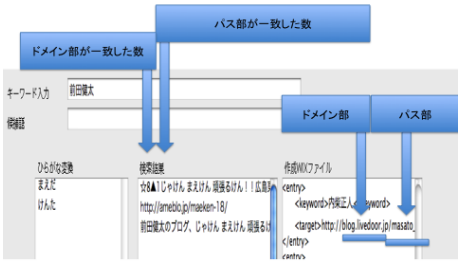

図 5 web 検索とターゲットのドメイン部とパス部の比較

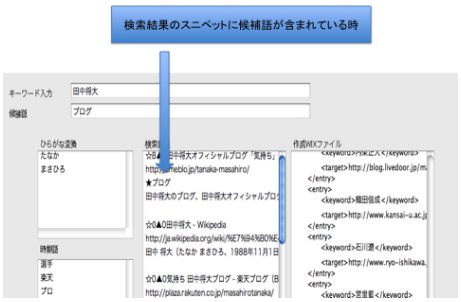

図 6　web 検索結果を形態素解析し、ターゲットに成りやすい語の表示

4.3　提案した手動記述システムの内部仕様

実装は Java のアプレットを用いた. 情報を取得するのに Yahoo ディベロッパーネットワーク開発と kizAPI(きざっぴ) を使用した.

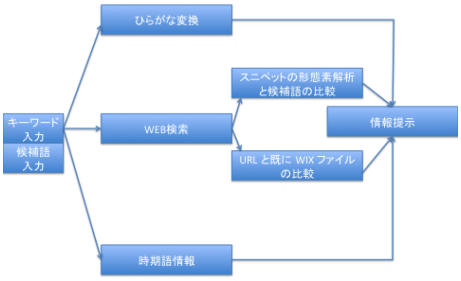

図 7　内部使用の概要

（1） 作成者から入力されたキーワードを得る.

（2） 入力されたキーワードに対応するのひらがな情報を Yahoo デベロッパーネットワーク開発のテキスト解析を使い情報を取得し, ひらがな変換部に提示.

（3） 入力されたキーワードに対応する時期語情報を kizAPI(きざっぴ) を使い上位 10 語を候補てして取得し, 時期語部に提示.

（4） Yahoo ディベロッパーネットワークのウェブ検索機能を使い, 入力されたキーワードからタイトル, URL, スニペットを得る.

（5） 得られた URL と既に WIX ファイルに記されているターゲットのドメイン部とパス部を比較して, 類似度の高いものに目印を付ける.

（6） スニペットを形態素解析して, ターゲットに成りやすい語（候補語）を含むものに目印を付ける.

（7） 最後に検索結果に情報をソートする. パス部が一致する検索結果がある時, ドメイン部が一致する検索結果がある時, スニペットを形態素解析した結果ターゲットに成りやすい語を含むものを上位にソートして提示.

5.　Web のリンク集にもとづく WIX ファイルの生成システム

ここでは Web のリンク集を自動的に発見し，WIX ファイルの生成と更新を行う．既存のリンク集からの WIX ファイル生成手法 [3] は，指定された単一の Web ページからリンク集を抽出し，WIX ファイルを生成する手法であった．Web 上からデータレコードを抽出したり，表を抽出する手法は数多くあるが，WIX に適切なリンク集のみを抽出することができない．そこで, リンク集抽出の新手法を提案し，WIX ファイル生成者の負担を少なくし，WIX システムに必要な情報源となる WIX ファイルを提供する．

5.1　リンク集を自動的に発見して WIX ファイルを生成する手法の概要

インプットとしてリンク集がありそうな Web サイト URL をユーザーが指定する．サイト内の全ページを対象として生成フェーズに入る．生成フェーズでは，初めに各ページをそれぞれブロック分割をする．ブロックからリンク集を抽出し，WIX ファイルを生成する．次に抽出器決定フェーズに入る．ここでは WIX ファイルからノイズを除去し，Web ページ上の最終的な抽出領域を指定するフィルタを利用する．フィードバックにより，抽出器決定フェーズで抽出に使ったフィルタを更新する．最後に監視フェーズに入る．Web ページが更新・追加される場合，抽出器に基づきシステムが自動的に WIX ファイル抽出対象ページか判断し，更新・生成を行う．

5.2　生成フェーズ

5.2.1　ブロック分割

Web ページはメニューやサイトマップ・広告などのような箇所と，メインコンテンツとなる箇所に分けられることが多い．抽出するべきリンク集は，メニューや広告を排除し，メインコンテンツのみを抽出する必要がある. そこで其々を分離するためにブロック分割を行う.

まずページの初期分割ポイントを指定する．HTML 記述を木構造と見て，各深さ n におけるブロックレベル要素(注1)の個数 Bn と割合 Rn を計算する．Bn が MAXBLOCK(注2)を超えた時点で探索をやめ，Rn が最大となる深さ n を初期の分割ポイントとする．

次にブロックレベル要素以下の内容をブロックとし，特徴ベクトルを計算する．ページ中におけるブロックの text，img，a，inline(注3)，sblock(注4)，dblock(注5)，script ノードの割合を計算し，ブロックの特徴ベクトル BV とする．各方向にリンク集らしさの補正(注6) をかけた重みつき特徴ベクトルを rBV とする．これらの特徴ベクトルから，類似度-占有度に基づいたメインブロックの決定と教師データを自動的に与えた半教師あり SVM によるメインブロックの決定を行う.

類似度-占有度手法では rBV の長さが最大のもの (rBVmax) からの比率が C1(注7)以上で，コサイン類似度が C2(注8)以上になるブロックをメインブロックとし，それぞれを再度ブロック分割する.

(注1)：div，p，table など
(注2)：初期値 10
(注3)：インライン要素 font，strong など
(注4)：ブロックレベル要素のうち単純な木構造となる p，pre，h1 など
(注5)：ブロックレベル要素のうち複雑な木構造を持つ teble，dl，div など
(注6)：実験でチューニングした値 (0.4，0.5，1.0，0.8，0.2，0.4，0.05)
(注7)：初期値 0.8
(注8)：初期値 0.7

半教師あり SVM では，rBVmin となる BV を負例，rBVmax となる BV を正例とした教師データを与え，残りを評価データとして判別を行う. その後，評価データだった BV のうち rBV が最小となるものを更に負例として教師データに追加し，再学習する. 評価データが残っているか，前回学習との評価の遷移が正から負へなったものがある限り学習を続け，学習が中断したときの正例をメインブロックとする．各ブロックがこれ以上分割すると a ノードが 10 未満になるか dblock ノードが 1 未満になった場合を atomic とし分割を終了する.

5.2.2 WIX ファイル生成

最終的に分割されたブロックからリンク集を抽出する．内部リンクとなっているアンカータグ集合を発見し，リンク先のパスの編集距離が 2 以下でアンカーが 10 個以上となる集合をリンク集とする.

それぞれのメイントピック内にリンク集構造があるかどうかを判断する．リンク集の発見は，アンカータグを含む頻出パターンを発見し，その集合を抽出することで行う.

- アンカータグ A1 を発見する
- 次のアンカータグ A2 を発見し，共有する最深の親 P(A1, A2) を導く
- それぞれのアンカータグにおける P(A1，A2) 以下の最小木を決める
- それぞれの最小木の編集距離が指定した閾値 t 以下であれば P(A1，A2) 以下のリンク集構造とする
- 次のアンカータグ A3 を発見し
  - P(A2，A3) が P(A1，A2) と等しいならば同様にして全ての最小木の編集距離を計算する
  - P(A2，A3) が P(A1，A2) と等しくなければ，P(A2, A3) 以下の最小木を新たに決め，編集距離を計算する
- リンク集構造と判定された最後のアンカータグの次のアンカータグを新たに A1 とし次のリンク集構造を探す

リンク集を抽出できたら，そこから WIX ファイルを生成する．リンク集のアンカーテキストの詳細情報を，ハイパーリンク先が記述していると考えられるので，アンカーテキストを keyword 要素に，リンク先を target 要素に格納し，1 リンクを 1 エントリーとする.

HTML 上のリンク集例

```
<html>...<table>...+
<a href="ichiro.html">イチロー</a>...
<a href="matsui.html">松井</a>...
...</table>...</html>
```

5.3 抽出器決定フェーズ

生成された WIX ファイルからデータ領域を指定する. このフィードバックにより抽出器に Keyword フィルタ・Target フィルタを追加する．メインブロックの指定は Xpath で記述し，WIX ファイル同士の統合を行う場合は SVM を指定して分離面の次元とベクトルを記述する．メインブロック指定の記述は以下のようになる.

WIX ファイル生成例

```
<wix><body><entry>
<keyword>イチロー</keyword>
<target>http://espn.com/ichiro.html</target>
</entry><entry>
<keyword>松井</keyword>
<target>http://espn.com/matsui.html</target>
</entry>...</wix>
```

/html/body/*[self::div or self::table or self::p or self::h1 or self::form][position()=3] /*[self::div or self::table or self::p or self::h1 or self::form][position()=1]

以上を元に指定したサイトの抽出器を決定する.

5.4 監視フェーズ

指定したサイトをクローリングし，ページが更新されたり追加された場合，抽出器に基づいて WIX ファイルを生成・更新する.

6. 評　　価

6.1 手動記述の評価実験方法

提案手法の優位性を評価するために, 以下の手順で評価実験をおこなった. 作成する WIX ファイルのエントリ数を同じ数（今回は 30 エントリとした）にすることで, 本手法の有無による作成時間の比較をおこなう.

（1） 一般のユーザー 5 名に WIX ファイルの記述の仕方を一読させ, 最低限の WIX ファイルの知識を得させる.

（2） 今回の手法を使わずに手動記述で WEB 検索を使い, 各ユーザに好みの WIX ファイルを作成した. そのとき, WIX ファイルのターゲットのカテゴリが同じになるように公式ページならば公式ページ, ブログならばブログ, 写真ならば写真に関するターゲットを選んで貰った. 記載するキーワード３０個は予め作成者に決めておいておかせる.

（3） 次に今回の手法を用いて全く同じ WIX ファイルになるようにエントリ数が 30 個の WIX ファイルを作成した.

- 評価方法 1:WIX ファイル作成時間の比較

上記の手順 2 で作成する WIX ファイルと手順 3 で作成する WIX ファイルのエントリ数ごとの作成時間を比較する.

- 評価方法 2:ターゲットとの適合率

情報検索システムの検索性能は主に正確性と網羅性の質的な観点から適合率 (precision:精度ともいう) を, 処理性能の量的な観点からスループットを測定することにより判定する.

上記の手順 2 で得られた WIX ファイルを正解 WIX とし, 上記の手順 3 で提示されたのターゲット候補がどのくらい正解 WIX と適合していたのかを評価する. そのとき, エントリの 10 個目まで, 20 個目まで, 30 個目まで推薦されたターゲットが正解 WIX のターゲット部にどのくらい適合しているかをはかる.

6.2 評価方法 1 の考察

この結果, 30 エントリの WIX ファイルの作成時間は本手法の有無により平均 74.2 秒短く成る結果となった.

エントリ数が増加するほど, 再現率が上がりターゲットを選ぶ

時間が減るため作成時間が短くなっていることが分かる.

6.3 評価方法 2 の考察

この結果より, エントリ数が多くなればなるほど本手法で提示するターゲットの適合率が上がる. エントリが増加すれば提示するターゲットの数が増えるため, 適合率が上がると考えられる. 但し, ターゲットカテゴリが同一の WIX ファイルを作成している場合推薦されるターゲット候補は 5 個程度 (※スニペットを形態素解析して、ターゲットに成りやすい語を含む場合は大きく増えることもある) である. そのため, さほど作成時間には影響しないかった.

6.4 評価方法 1 と評価方法 2 からの考察

適合率の上昇とともに作成時間の削減にも繋がっていることが分かる. このことからこのシステムはある程度大きな WIX ファイルの作成に適していることが分かる.

6.5 一番上位に推薦されたターゲット候補がターゲットに成らなかった理由

ターゲット候補の上位に推薦されながら正解に成らなかったケースがある.

例えば, 野球選手ブログ WIX の場合, 大手ブログサイトを利用している選手のターゲットの中にその他のブログサイトを利用している選手を入力すると正しくない他人の書いたページ等を推薦してしまう. 新たに記載するエントリのドメイン部が既に記されている WIX ファイルのエントリのドメイン部にない時, 上位に推薦されない.

6.6 Web のリンク集にもとづく WIX ファイルの生成システムの評価方法

BV の重み付けと閾値の調整し, 本手法の有用性を再現率, 適合率を用いて判断する.

6.7 占有度-類似度抽出の閾値決定

BV の重み付けと閾値の調整をするために，5 つのサイト(注9)で実験を行った. 各サイトにおいて，リンク集を抽出されるべきではないページと WIX ファイルに適正なページをそれぞれ 10 ページずつ指定し，生成された WIX ファイル自体の適合率 File-Pre・再現率 File-Rec と各エントリーの適合率 Node-Pre・再現率 Node-Rec を測り，再現率が 100%になり，適合率が高くなるようにチューニングを行った. その結果を表 1 に示す. File Pre(FP) はファイル適合率，File Rec(FR) はファイル再現率，Node Pre(NP) はエントリー適合率，Node Rec(NR) はエントリー再現率として表記する.

表 1 各サイトから抽出した WIX の精度

サイト	File Pre	File Rec	Node Pre	Node Rec
Soccernet	71	100	87	100
Espn	77	100	82	100
Sony	71	100	83	100
Kakaku	83	100	98	100
Tokyu	91	100	99	100

(注9)：http://soccernet.espn.go.com/ http://espn.go.com/ http://www.sony.jp/ http://kakaku.com/pc/desktop-pc/ http://www.tokyu.co.jp/

6.8 占有度-類似度抽出と半教師 SVM の比較

次に，占有度-類似度手法 (1) と半教師あり SVM(2) で再現率と適合率を比較した実験を表 2 に示す. チューニングに使った 2 つのサイトの 20 ページと，日本野球選手の情報サイトの指定ディレクトリ以下 2120 ページ，4 字熟語サイトの全 2700 ページを対象に行った.

どちらの手法でも適合率があまり高くないが，現状では簡単なフィルタリングのみしか行っていないため，文字列の長さや瀕死，上層へのリンクなどのフィルタを充実させる必要がある. また適合率は WIX ファイルのみの問題である.

一方再現率はチューニングしたサイト以外でもほぼ 100%になっている. WIX ファイルを抽出する際にエントリーやファイルとしての見落としがあるとその WIX が情報を網羅出来ていないことになり，価値が下がると言える. また WIX ファイルと抽出に指定したサイトと両方の問題になる.

表 2 手法 1 と 2 の比較

サイト	FP1	FP2	FR1	FR2	NP1	NP2	NR1	NR2
Soccernet	71	71	100	100	87	82	100	100
Kakaku	83	100	83	100	98	89	100	100
nbp	76	72	100	100	81	78	100	100
4jinavi	82	83	89	93	85	83	98	99

6.9 複数の WIX ファイルの記述

手動で WIX ファイルを記述する場合と WIX ファイル化対象ページを手動で指定する半自動方法，全自動手法 (占有類似, SVM) の 4 つの比較を表 3 に示す. 提案手法のうち，占有度-類似度に注目した手法では，サイトの URL を指定するだけでリンク集を発見し，該当するページの数だけ WIX ファイルが自動的に生成される. 半教師 SVM ではさらに，WIX ファイルの抽出に使ったブロックベクトルの分離面を元に WIX ファイルをクラスタリングし，自動的に統合することが可能となる.

表 3 各手法における WIX 生成の適性

	手動	半自動	自動: 占有類似度	自動: SVM
適合率	○	○	△	△
再現率	△	○	○	○
自動統合	—	×	×	○
手動時間	×	△	○	○
抽出時間	—	○	△	×
更新	×	△	○	○

7. 結　　論

本研究では手動記述支援システムと Web のリンク集にもとづく WIX ファイルの生成システムを提案し実現した. 手動記述支援システムにより, 30 エントリの WIX ファイルの作成時間を平均 74.2 秒 短くし, 情報提示の精度は 80 %になった。Web のリンク集にもとづく WIX ファイルの生成システムにより, 再現率を 100 できることが確認できた. また, 既存手法や手動記述に対して比較を行い, 本研究によって大量の WIX ファイルを生成することが可能となった.

文　　献

[1] 佐藤 裕紀, 遠山元道, “ A-doc ファイルのアタッチの機能を持つ専用ブラウザの試作 ”, データ工学ワークショップ, *DEWS2007*, 2007

[2] 高橋 健太郎, 遠山元道, “ SuperSQL による A-doc ファイルの生成“ , データ工学ワークショップ, *DEWS2007*, 2007

[3] 市東 隼, 遠山 元道: 『A-doc に基づく WEB リンク集再利用化ツールの試作』, 電子情報通信学会第 *19* 回データ工学ワークショップ, 第 *6* 回日本データベース学会年次大会論文集, 2008.