

Web Index システムにおけるコンテンツ作成支援

Content Creation Support System of Web Index

藤井 洋太郎[▼] 遠山 元道[◆]

Yotaro FUJII Motomichi TOYAMA

著者らは、Web ドキュメント中に存在する単語に対してリンクを生成する Web Index(WIX)システムを開発している。リンク生成には、キーワードとそれに対応する URL のエントリーが記述された WIX ファイルを事前に作成する必要がある。本稿では、XPath を利用し閲覧 Web ページのエントリー抽出・ファイル自動生成を可能にするウェブブラウザ拡張機能と、オンラインでのファイル作成・編集を可能にするファイル編集システムの 2 つのアプローチを提案し、実装した。

Web Index (WIX in short) system joins web pages by hyperlink. WIX file is a XML file, having entries consisting of a keyword and the URL corresponding to it. When a user joins (attaches) WIX files on a web page, keywords on the web page are changed into hyperlink to the URL. In this study, we proposed two approaches for creating and editing WIX file. First system, Using XPath, enable to automatically generate the WIX file from Web pages viewed now. Second system allows creating and editing WIX files in the online.

1. はじめに

近年、Web2.0の広がりによって多くのユーザが集合知を形成し、利用するといったWebの利用形態が一般化してきた。その代表例であるWikipediaでは記事中の単語がそれに対応する記事へリンクされており、ユーザの検索負担が軽減されている。また、Rada Mihalceaら[1]やDavid Milneら[2]によって任意のWeb文書のキーワードにWikipedia記事へのリンクを自動生成するWikificationシステムが提案されている。しかし、これらのシステムにおいて生成されるリンクはWikipedia記事へのリンクのみであり、時と場合によりニュース記事や公式ブログ等、異なる情報を必要とする場合があると推測される。このため、これらのシステムは必ずしもユーザのニーズに応じた情報を結合し、提供出来ているとは言えない。

一方、著者らが研究を行なっているWeb Index(WIX)では関係モデルの考え方をWebの世界に持ち込むことによって、ドメインにとらわれない、ユーザ主体のWeb情報資源の結合を図っている[3]。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キーワードを対応するURLのハイパーリンクに自動的に変換する(図1)。このようにWIXでは単語1つに対して様々なURLへのハイパーリンクを生成することが可能である。

リンクの生成にはWIXファイルが必要となるが、希望のWIXファイルがない場合、別途WIXファイルの作成が必要となる。作成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[4]。手動での記述の場合、手間と時間のコストが非常に大きい。一方、Webページのリンク集などからの自動生成も考えられるが、プログラミングなどの知識がない一般的ユーザにとっては敷居が高い方法である。また、作成済みのWIXファイルに対して、リンク先の変更や追加などの情報の更新が必要になった場合、即座に反映することが難しい。そこで本論文では、WIXファイル作成者が簡単にWIXファイルを自動生成できるシステムと、オンラインでWIXファイルの作成・編集を支援し情報の変更即座に対応できるシステムが必要と考え、以上の2つのアプローチによるWIXファイル生成の効率化を目的とした。

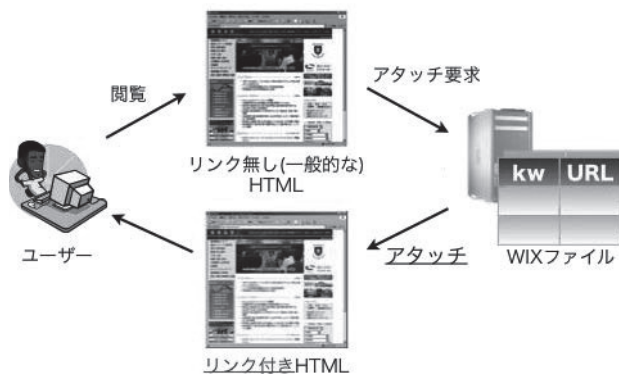


図1 Web Index システム

Fig.1 Web Index System

2. Web Index システム (WIX)

2.1 WIX ファイルの書式

WIX ファイルは XML 形式で記述されたエントリーの集合であり、記述例は図2のようになる。エントリー内には、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。また、WIX ファイルには、header 要素にファイル概要、作者コメントなど、その WIX ファイル全体についてのメタデータを格納することも可能である。

2.2 アタッチ

閲覧中の Web 文書にある単語について詳細な情報を得たい場合、従来ならその単語を1つずつ検索エンジンなどで検索しなければならない。これに対し WIX では閲覧中の Web 文書に WIX ファイルを結合することで、キーワードからそのキーワードの情報へのハイパーリンクを生成し、Web サイトを動的に結合することが可能である。この結合操作を“アタッチ”と呼ぶ。また、同じキーワードでもリンク先の異なる WIX ファイルを複数提供している場合、利用者は状況に応じた WIX ファイルを選択・アタッチすることで、より目的に適した情報を得ることが可能となっている。

[▼] 学生会員 慶應義塾大学大学院理工学研究科開放環境科学専攻修士課程 1 年 yotaro@db.ics.keio.ac.jp

[◆] 正会員 慶應義塾大学理工学部情報工学科准教授 toyama@ics.keio.ac.jp

～企業製品 WIX の例～

```
<wix>
  <header>
    <description> アップル製品情報 </description>
  </header>
  <body>
    <entry>
      <keyword>iPhone</keyword>
      <target>http://www.apple.com/jp/iphone/</target>
    </entry>
    <entry>
      <keyword>iPad</keyword>
      <target>http://www.apple.com/jp/ipad/</target>
    </entry>
  </body>
</wix>
```

図 2 WIX ファイルの記述例

Fig.2 Example of the WIX File

2.4 WIX ライブラリ

全ての WIX ファイルは、WIX ライブラリ(本大学で運営予定)で管理され、利用者が自由に WIX ファイルを作成・公開でき、また他ユーザが作成した WIX ファイルを利用することができる。希望の WIX ファイルが公開・作成されていない場合、別途 WIX ファイルを作成しライブラリに登録する必要がある。

2.3 WIX ファイルの性質

あるキーワードに対してターゲットになりうる URL は Web 上に複数存在する。例えば、キーワードがサッカー選手や芸能人の氏名である場合、公式サイト、ブログ、Wikipedia などのリンク先が候補として挙げられる。ターゲットは WIX ファイル作成者が WIX ファイルの目的に適したものを選択し記述する。ここで、一つの WIX ファイル内のキーワードとターゲット先は共通点があるものを記述する必要がある。仮に、キーワードによってターゲット先が Wikipedia であったり公式サイトであったりと異なるカテゴリ先が記述された WIX ファイルの場合、利用者の目的に応じたアタッチは困難となる。そのため、図 3 の右例のように同一ファイルの 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。

キーワード : 本田圭佑 ターゲット先 : 公式サイト	キーワード : 本田圭佑 ターゲット先 : Wikipedia
キーワード : 長友佑都 ターゲット先 : 公式ブログ	キーワード : 長友佑都 ターゲット先 : Wikipedia

不適切な WIX ファイル例

適切な WIX ファイル例

図 3 適切な WIX ファイル

Fig.3 An appropriate WIX File

3. WIX File Extractor : ウェブブラウザ拡張機能を用いた WIX ファイル生成

3.1 リンク集

Web には様々なリンクを持つページがあるが、その中には共通の意味を持つリンク集を含むページが存在する。そのようなリンク集を持つページの例を図 4 に示す¹。図 4 にお

る共通の意味を持つリンク集とは、FC バルセロナの選手名である。このリンク集において、アンカーテキストである選手名を keyword に、リンク先を target に記述することで選手情報へのリンクをまとめた WIX ファイルを作成することができる。

このとき目標とするリンク集は選手名だけであるが、上位ディレクトリへのリンクや別のコンテンツへのリンクなど、目標としないリンクも含まれる場合がある。Web からリンク集を発見し、さらに WIX ファイルのエントリーとならないリンクを除去して WIX を生成することが必要である。



図 4 リンク集の例

Fig.4 Example of Links

3.2 システム概要

本システムでは、ウェブブラウザの拡張機能を用いて、Web ページのリンク集から WIX エントリーを抽出し、自動的に WIX ファイルを作成することを可能にする。

ユーザは目的のリンク集がある Web ページを表示し、ページ上をマウスオーバーすることで、視覚的にリンク集を発見できる。また、目的のリンク集が見つかったときは、ボタンひとつで WIX ファイルを確認・生成することができる。図 5 はこのシステムを用いてサッカーチームの選手一覧ページから、WIX ファイルを生成する例である。

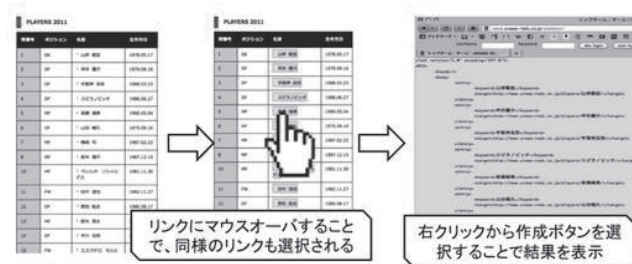


図 5 WIX File Extractor 動作例

Fig.5 Example of the Use of the WIX File Extractor

3.3 システム外部仕様

本システムの利用の流れは以下の通りである。主に以下の 4 つのフェーズを行うことで、WIX ファイルを生成できる。

(1) WEB ページ上をマウスオーバー。

目的のリンク集がある Web サイト上を読み込み、ペー

¹ <http://www.fcbarcelona.jp/team/2011/12/>, 参考・抜粋, 2012 年 5 月 17

日アクセス。

ジ上をマウスオーバーする。

(2) 目的のリンク集を探す。

目的のリンク集が選択される場所を探す。

(3) 右クリックからファイル生成ボタンを押す。

目的のリンク集が選択されたら、右クリックメニューから生成ボタンを押す。

(4) 表示された結果を確認。

また、オプションの機能として、選択範囲の広狭や取得するキーワード内の空白の有無などの指定が可能。

3.4 システム内部仕様

本システムは、ウェブブラウザ(Firefox, safari, Google Chrome)の拡張機能として実装した。本研究では目的のページからリンク集を認識するためにXPathを利用し、目的の情報を抽出する[5, 6]。ユーザがマウスオーバーしているノードを取得し、HTMLソースのbodyノードまでのXPath式を作成することで、ユーザに対象ノードと同様の構造をとっている兄弟ノードの集合をユーザにフィードバックを与えることができる。処理の流れは以下の通りである。

(a) XPath 式の作成

システムはユーザのマウス操作を監視し、マウスオーバーされているノードを取得する。取得したノードからbodyノードまでのXPath式を作成する。XPath式を作成する際、それぞれのノードのclass属性、id属性も同時に取得する。XPath式にそれらの情報を指定することで、ノードまでのより正確なXPath式を作成できノイズの除去ができると期待できる。一方、目的のリンク集を指定できない場合は、XPath式から属性の指定を除き、範囲の指定を緩めることができる。以下の図6は、図5においてマウスオーバーされたノードから得られるXPath式の例である。

```
//BODY/DIV[@class="container"]/DIV[@class="innerBox"]/DIV[@class="mainColumn"]/  
DIV[@class="mainContentColumn"]/TABLE[@class="mainContTblA mgB20 teamList"]/  
TBODY/TR/TD/UL[@class="mainContList4"]/LI/A
```

図6 XPath式の例

Fig.6 Example of the XPath Formula (Fig.5)

(b) ユーザにフィードバック(ノードの色付け)

作成したXPath式に適合する全ての兄弟ノードを色付けする。ユーザはフィードバックを受けながら、目的のリンク集が得られるリンク集を探す。

(c) エントリーの抽出

ファイル生成ボタンが押されると、そのXPath式からキーワードとターゲットの抽出を行う。

(1) キーワードの抽出：キーワードはアンカータグに囲まれたアンカータグから取得する。しかし、リンクがテキストでなく画像である場合、アンカーテキストが存在しない可能性もあり得る。そこでリンクが画像の場合は、IMGタグ内のalt属性などから取得する。また、ユーザの設定により、キーワード内のスペースの有無を指定できる。

(2) ターゲットURLの抽出：ターゲットはアンカータグのhref属性から取得する。

(d) 抽出結果の表示

得られたキーワードとターゲットを、2.1のWIXファイルの書式に整形し、ユーザが閲覧しているページ上にWIXファイルの内容を表示する。

4. Online WIX File Editor : オンラインでのWIX ファイル作成・編集

4.1 背景

これまでのWIXシステムでは、WIXファイルの作成・編集はファイル作成者がそれぞれのローカル上でファイルを作成し、システムにアップロードするという方法をとっていた。しかし、WIXファイル内のターゲット先の変更や、エントリーの追加や削除の更新を行うたびにファイルを編集し、アップロードするのは、非常に高コストである。また、一人の作成者によって、数百～数千のエントリーを手動で記述することは現実的に難しいと考えられる。

そこで、WIXファイル作成支援の2つめの方法として、WIXファイルをオンラインで編集可能にするシステムを提案する。

4.2 システム概要

本章では作成者が簡単にWIXファイルを閲覧でき、エントリーの追加・削除などの更新を行えるオンライン編集システムの構築を行った。本システムでは、主に以下の作成支援機能を実装した。

- ・新規エントリーの追加・削除
- ・WIXファイルのヘッダー情報追加
- ・キーワードに対するターゲットの推薦
- ・WIXファイル内エントリーの検索
- ・編集履歴閲覧・編集取消
- ・登録済みエントリーのリンク切れチェック
- ・新規WIXファイル作成

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。

4.3 システム外部仕様

図7に、以下の機能を実装したシステムの外観を示す。

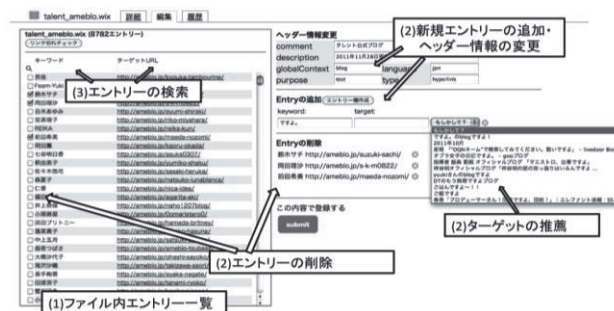


図7 Online WIX File Editor : 外観

Fig.7 Online WIX File Editor : Appearance

本システムの利用の流れは以下の通りである。

(1) WIXファイルの選択

WIXファイル一覧から、閲覧・編集したいファイルを選択。

(2) WIXファイルの編集

エントリーの追加や、ヘッダー情報は所定のフォームに入力する。またエントリー追加の際、キーワードを入力するとそのキーワードに対するターゲットの候補が自動的に表示される。ユーザはその候補を選択することでも、ターゲットの入力ができる。エントリーの削除は、ページ左のエントリー欄のチェックボタンを入力することで削除できる。

(3) エントリーの検索

エントリー数が多く、目的のエントリーを探せない場合は検索フォームにクエリを入力することで、適合する全てのエントリーを表示できる。検索結果はクエリを入力するたびに表示されるようになっている。図8にエントリー検索機能を使った例を示す。

(4) 履歴の閲覧・編集取り消し

全ての編集は履歴として保存され、編集の取り消しが可能となっている。

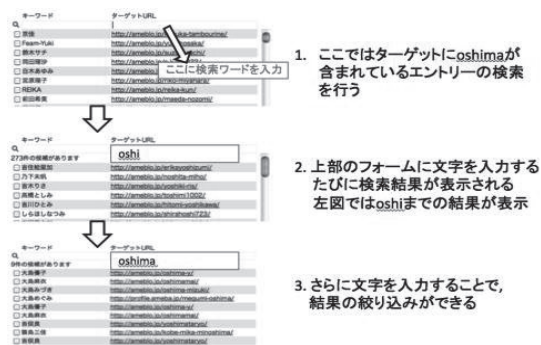


図8 エントリーの逐次検索

Fig.8 Incremental Search of WIX Entry

4.4 システム内部仕様

4.4.1 WIX ファイルのロード

XML 文書を読み込み操作する方法として、主に DOM を利用する方法と SAX を利用する方法の2種類の方法が挙げられるが、WIX ファイルのエントリー数は対象が Wikipedia などであると数 10 万件にも及ぶ可能性がある。そこで、メモリの消費量が少なく、高速に文書を走査できる SAX パーサーを利用し WIX ファイルをロードする。

4.4.2 エントリーの逐次検索

検索フォームにキーが押されるたびに、システムは Ajax 通信を行いファイル内のエントリーと入力クエリ内容が部分一致するエントリーを検索し検索結果を表示する。Ajax 通信を用いた非同期通信を行うことでページ遷移をすることなく、動的に検索結果の書き換えを行う。

4.4.3 ターゲットの推薦

エントリー追加フォームにキーワードが入力されると、バックグラウンド上でそのキーワードに対する検索エンジンからの検索結果を取得する。得られた検索結果の HTML ソースを解析し、タイトルと URL を取得し、その上位 k 件 (k は任意) の結果を提示する。また WIX ファイルのヘッダー情報には、WIX ファイル全体についてのコンテキストを格納できるグローバルコンテキスト属性 (ex. サッカー WIX であれば、“サッカー”。公式ブログ WIX であれば、“ブログ”等) があり、WIX ファイルがその属性を持っている場合、キーワードとグローバルコンテキストの AND 検索を行った結果を提示する。グローバルコンテキストを加えることで、より作成者の目的にあったターゲットを提示できる。提示した結果をユーザが選択すると、その URL をターゲットフォームに自動的に補完する処理を行う。

4.4.4 編集履歴

編集の際、以前のバージョンとの差分を取得し、その差分を用いることで、以前の版に戻すことを可能にした。

5. 評価

以上の 2 つのシステムの有効性を評価するため、各システムを利用したユーザ実験を行い、ファイル作成時間と精度について評価する。実験は 3 名の HTML や XML を記述した経験のある被験者 (以下、経験者) と、3 名の HTML を記述した経験のない被験者 (以下、未経験者) の 6 名に参加してもらい、以下の 2 つの実験を行い評価を行った。

5.1 WIX File Extractor の評価

まず、WIX File Extractor の評価を行う。

5.1.1 評価方法

本システムの有用性を評価するため、本システムを利用した場合と利用しない場合の WIX ファイルの作成時間、タイピング数 (キーボードタイプ数、クリック数の合計) の差を測定した。利用しない場合の時間とタイピング数は、事前に著者が 10, 20, ..., 100 件のエントリーを持つ WIX ファイルを作成し、それぞれの WIX ファイル作成にかかった時間、タイピング数の測定結果を用いた。被験者には、以下の 3 つのカテゴリの WIX ファイルを本システムを用いて作成してもらった。

- (a) あるスポーツチームの選手情報を対象にした WIX ファイル
 - (b) あるメーカーの製品情報を対象にした WIX ファイル
 - (c) 被験者が興味を持っているカテゴリの WIX ファイル
- また、その後 5 分間本システムを利用して自由に WIX ファイルを作成してもらい、それらの WIX ファイルの精度 (再現率、ノイズ数) を測定した。

5.1.2 作成時間とタイピング数

図 9 に、筆者が事前にエントリー数の異なる 10 個の WIX ファイルを作成した際の作成時間と、タイピング数を示したグラフである。図 10 は、本システムを利用し作成した WIX ファイルの作成時間とタイピング数を表したグラフである。

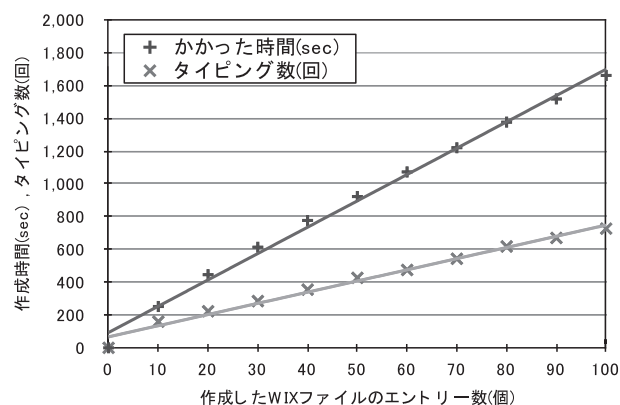


図9 本システム未使用時

Fig.9 Without Use of WIX File Extractor

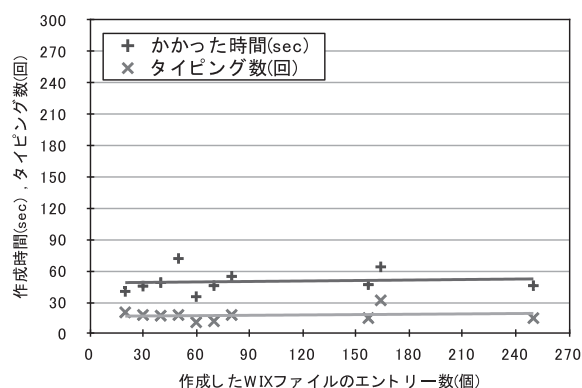
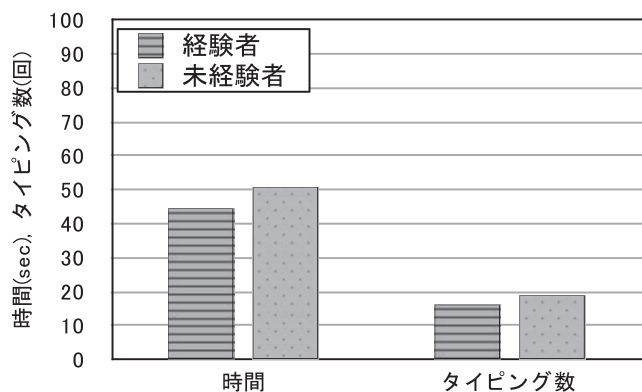


図 10 本システム使用時

Fig.10 Use of WIX File Extractor

まず、図9のグラフより、作成するWIXファイルのエントリー数が増加するにつれ、作成時間、タイピング数も線形的に増加していることが分かる。一方、本システムを利用して得られたWIXファイルはエントリー数にかかわらず、ほぼ一定の時間とタイピング数で作成されている。このことから、大規模なWIXファイルを作成する際に本システムを利用することで、大幅なコスト削減ができると考えられる。また、図11に経験者と未経験者ごとの平均作成時間と平均タイピング数を示す。ここから読み取れるように両者の差は作成時間で約5秒、タイピング数で約5回と僅かなものであり、プログラミングに詳しくない一般的なユーザでも利用可能であるといえる。



WIXファイル生成の平均時間と平均タイピング数

図 11 経験者、未経験者ごとの作成時間とタイピング数

Fig.11 Difference by The Experience Presence

5.1.3 生成 WIX ファイルの精度

次に本システムによって作成されたWIXファイルの精度の評価を行う。評価の対象となるWIXファイルは全36種類であり、全ファイルにおける再現率とノイズの個数を測定した。再現率は(1)式のように求めた。

$$\text{再現率} = \frac{\text{期待通り抽出できたエントリーの総数}}{\text{抽出したいWIXエントリーの総数}} \times 100 \quad (1)$$

表1は全ファイルの再現率ごとに分類した表である。この表より約7割のWIXファイルで全ての目的のエントリーを得られたことがわかる。また、約9割のWIXファイルが再現率

80%以上の割合で生成できており、本システムは非常に有用であるといえる。一方再現率が0%となり目的のエントリーが1つも得られないWIXファイルも存在した。これらの原因として、Webサイトにおけるアンカーテキストの記述と、サイト特有のHTML構造が原因としてあげられる。本システムは対象Webサイトの構造に大きく依存しているため、アンカーテキストに目的以外のワードが入っていたり、全てのノードを参照してしまうようなHTML構造には対応できなくなってしまう。今後、再現率や適合率をさらに高めるためには、アンカーテキストの部分指定や、XPath式の範囲指定を可能にするアルゴリズムを考える必要がある。

表 1 生成した WIX ファイルの再現率ごとの分布

Table 1 Recall of Created WIX File

本システムによる再現率							
再現率(%)	0	10~60	70	80	90	100	計
ファイル数(個)	2	0	2	1	6	25	36

5.2 Online WIX File Editor の評価

次に Online WIX File Editor の評価を行う。

5.2.1 評価方法

本実験では、被験者に既にシステムに登録されているWIXファイルの編集を行ってもらった。以下の3つWIXファイルを対象としている。

- (a) 企業サイト WIX ファイル(約 100 エントリー)
- (b) タレントブログ WIX ファイル(約 1 万エントリー)
- (c) Wikipedia WIX ファイル(約 10 万エントリー)

編集内容は、著者がランダムに指定した10個のエントリーの追加、ランダムに指定した10個のエントリーの削除の2通りを行なった。本実験は被験者にシステムを利用する場合と、利用しない場合の2通りの作業を行なってもらい、編集時間の差を測定した。本実験での編集時間とは、システムを利用する場合は本システムにアクセスした瞬間、利用しない場合は対象WIXファイルのダウンロードをはじめから再度システムにアップロードし終わるまでの時間とした。

5.2.2 編集時間の評価

まず、本システムの使用有無による、編集時間を図12に示した。縦軸に編集時間、横軸はWIXファイルのエントリー数順にとった。図12から読み取れるように、全ての場合において本システムを利用したほうが良い結果となった。システムを利用しない場合の編集時間はエントリー数が大きくなるにつれ、大きくなっていることがわかる。この問題の大きな原因として、エントリーの増加に伴い類似するエントリーが増え、検索するコストが大きくなってしまうことがあげられる。一般的なエディタの検索機能では結果を一度に把握するのが難しいが、本システムでの検索は、1文字ごとに検索結果が変化し直感的な操作をすることが可能である。実際、本システムでのエントリー数による編集速度の変化は小さいものであった。またターゲット推薦などの機能などにより編集における検索コストなどの削減ができ、本システムは有用性があるといえる。

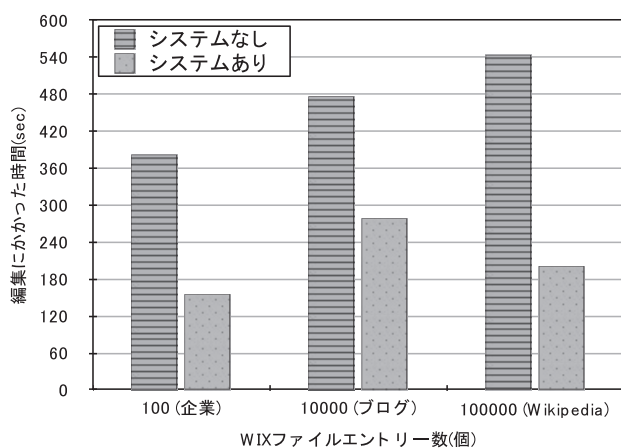


図 12 WIX ファイルの編集時間

Fig.12 The Time Taken to Edited WIX File

5.2.3 経験の有無による差

次に本システムを利用した際の経験者と未経験者の作成時間の差を評価する。図 13 は、カテゴリごとに経験者と未経験者の作成時間(sec)の平均を表したものである。どの WIX のファイルの編集の場合でも、経験者のほうが速かったが、いずれの場合においても、その差はわずか 30 秒ほどであった。本システムは、XML などの特別な知識を必要とせず、直感的な操作で編集ができるため、どのユーザでも同程度の作成時間で編集を行えたと考えられる。

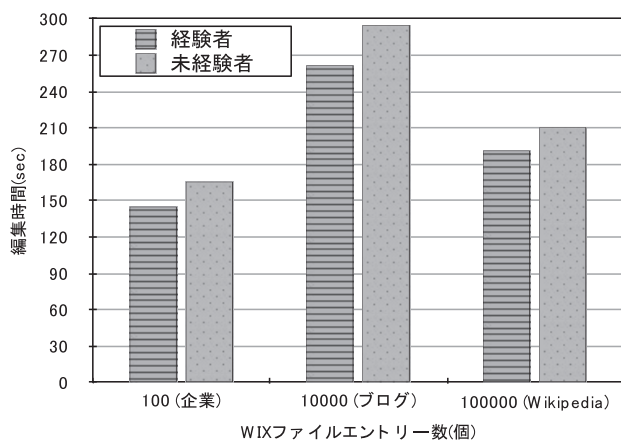


図 13 経験の有無による編集時間の差

Fig.13 Difference by The Experience Presence

6. まとめと今後の課題

本論文では、2つのアプローチによる WIX ファイルの作成支援システムを提案し実装した。

1つ目に提案した、WIX File Extractor を利用することで、作成者のファイル作成にかかるコストを大幅に減少させることができた。また、HTML や XML などのプログラミング知識が無いユーザでも目的の WIX ファイルを作成することが可能であった。どのユーザでも簡単に素早くコンテンツ作成ができることや、複数のウェブブラウザの拡張機能として本システムを利用することができることは大きな利点であると考えられる。

また、2つ目に提案した Online WIX File Editor を利用することで、作成者は即座に必要な情報の更新が行え、また直感的な操作により XML などの知識がないユーザでも簡単に目的の WIX ファイルの作成・編集が可能であった。また、オンライン上に WIX ファイルを公開することで、一つの WIX ファイルを複数のユーザで編集することが可能になり、WIX ファイル作成の効率化も期待できる。

これらのシステムを提供することで、一般的なユーザでも簡単に WIX ファイルを作成し編集することが可能となる一方、悪意あるユーザによるあらし行為や攻撃などが増加することが考えられる。そこで今後は、認証システムやブラックリスト、WIX ファイルやユーザの評価などの対策をおこない、ユーザが安心して WIX システムを利用できる環境づくりの構築を行なっていく。

【謝辞】

これまで WIX の研究・開発に従事してくださった皆様に感謝する。

【文献】

- [1] Rada Mihalcea and Andras Csomai.: "Wikify!: Linking documents to encyclopedic knowledge", In Proceedings of the 16th ACM Conference on Information and Knowledge Management, pp.233-242, 2007.
- [2] David Milne and Ian H. Witten.: "Learning to link with Wikipedia", In Proceedings of the 17th ACM Conference on Information and Knowledge Management, pp.509-518, 2008.
- [3] 森 良介, 藪 達也, 朱 成敏, 遠山元道. Keio WIX システム(2) サーバーサイド実装, DEIM2011, 2011.
- [4] 市東 隼, 分部 亮太, 朱 成敏, 遠山元道. Keio WIX システム(3) コンテンツ作成, DEIM2011, 2011.
- [5] Yanhong Zhai and Bing Liu.: "Structured Data Extraction from the Web based on Partial Tree Alignment", In IEEE Transactions on Knowledge and Data Engineering, pp.1614-1628(2006).
- [6] 山本 岳洋, 中村 聡史, 田中 克己. RerankEverything: ランキング結果閲覧のための柔軟な再ランキングインタフェース, TOD, Vol.3, No.4, pp. 48-64, 2010.

藤井 洋太郎 Yotaro FUJII

慶應義塾大学大学院理工学研究科開放環境科学専攻修士課程在学中. 2012 慶應義塾大学理工学部情報工学科卒業. 情報処理学会学生会員. 日本データベース学会学生会員.

遠山 元道 Motomichi TOYAMA

慶應義塾大学理工学部情報工学科准教授. 昭和 54 年慶應義塾大学工学部管理工学科卒業. 昭和 59 年同大学大学院博士課程修了後, 管理工学科助手, 専任講師を経て現職. 博士(工学). 平 8 年 Oregon Graduate Institute 客員研究員. 平成 10~13 年科学技術振興事業団さきかけ研究 21

「情報と知」領域研究員. 主にデータベースの研究に従事. 電子情報通信学会, 日本データベース学会, ACM, IEEE 各会員.