F-007

編集距離を組み込んだ Wrapper による Web からの情報抽出

Information Extraction on the Web in Wrapper using Levenshtein Distance

坪島 恭平 †　　大和田 勇人 ††
Kyohei Tsuboshima　　Hayato Ohwada

1　はじめに

近年インターネットの発達によって Web 上には膨大な量の情報が存在するようになった．ユーザーはインターネットを通じて，Web から自由に情報を得られるようになった一方で，膨大な量の情報が存在するため，自分の欲しい情報のみを得るためには多くの手間と時間を要するようになった．しかし，Web 上に存在する文書の多くが HTML などの半構造化文書であり，ブラウザ閲覧を前提にしていることから，Web ページから情報を取得して容易に利用することが困難である．そのため，このような膨大な量の情報を効率よく扱うために，HTML や XML などの半構造化文書から有用な情報を発見・抽出するための情報抽出技術が必要となっている．

そこで，本研究では情報抽出技術の 1 つである Wrapper [1] に着目した．Wrapper とは，HTML などで書かれた Web ページから特定の情報を自動的に抽出するための，サイトレイアウトを利用した抽出ルールおよび抽出プログラムのことである．よって，Wrapper を生成することでこれら半構造化文書から，レイアウト情報を利用して特定の情報の位置を把握し，必要な情報のみを抽出することが可能である．しかし，Wrapper はサイトレイアウトに依存するため，サイトレイアウトが異なるページに対して適用できないことから，サイトごとに手動で Wrapper を生成することは現実的ではなく，自動的に Wrapper を生成することが課題となる．そのため，抽出ルールを自動的に導出する手法が研究されている [2][3].

本研究では，HTML などの半構造化文書でつくられた Web ページから必要な情報のみを取り出す新しい手法を提案する．そのために本研究では，拡張ブラウザと編集距離を用いる．まず，拡張ブラウザ上においてユーザが求める情報を基に，その情報が含まれている Web ページを収集し，ページの構造解析を行う．そして，HTML パスの編集距離を算出し，構造上の類似性を求めることで類似性の高いものを抽出結果として取り出す．この手法によりユーザーが求める情報を選択することで，Web ページのソース上でその情報と類似した構造を持つ同様な情報を自動的に抽出することができる．また，この提案手法の有効性を実証すべく，提案手法を用いたシステムを実装し，評価実験を行う．

† 東京理科大学大学院理工学研究科経営工学専攻，Department of Industrial Administration, Graduate school of Science and Technology,Tokyo University of Science
†† 東京理科大学理工学部経営工学科, Department of Industrial Administration, Faculty of Science and Technology, Tokyo University of Science

2　提案手法

2.1　提案手法の概要

本研究では，半構造化文書から特定の情報のみを抽出するために拡張ブラウザと編集距離を用いた手法を提案する．提案手法の概要を図 1 に示す.

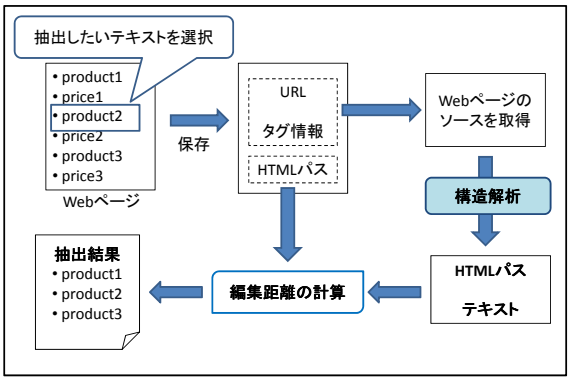

図 1　提案手法の概要

本手法はユーザーが選択したテキスト情報を入力として，編集距離を用いて構造的に類似したテキストを出力する．その流れを以下に示す.

1. Web ページから求める情報を選択する．この際にページの URL，選択したテキスト，そのタグ情報と HTML パスを取得する
2. 取得した URL を基にページのソースコードを取得し，ページの構造解析を行う
3. タグ情報を基にソース全体から選択したテキストと同じタグに属するテキストとその HTML パスを取得する
4. 選択したテキストの HTML パスと同じタグに属する HTML パスを編集距離を用いて類似度を求める
5. 編集距離の値が小さいければ類似度が高いことから，編集距離の小さいものを抽出結果として出力する

以下に本手法で用いる拡張ブラウザと編集距離について述べる.

2.2　拡張ブラウザ

拡張ブラウザを用いて情報抽出するためのユーザーインターフェースを実装した．インタフェース用のブラウザには拡張性の高い Firefox を用い，Firefox アドオンである Greasemonkey[4] を用いて，JavaScript で記述したユーザスクリプトを実行することにより実現した.

このユーザインターフェースにおいて実装した機能は，抽出可能なテキストのハイライトおよび抽出項目の保存である．具体的には Web ページ上において，抽出可能なテキストにマウスをのせると背景色が変わる機能，選択後のテキストを青枠で囲む機能，Ctrl キーと Shift キーの同時押下時に抽出項目に関する情報を保存できる機能を実装した．

2.3 編集距離

この手法ではページの構造に着目した抽出として，編集距離を用いてテキストのソース上での位置を示す HTML パスを基に類似性をはかることで，似たような構造を持っているかを調べる．編集距離 (あるいはレーベンシュタイン距離) は，2 つの文字列がどの程度異なっているかを示す数値である．具体的には，文字の挿入や削除，置換によって，1 つの文字列を別の文字列に変形するのに必要な最小手順回数を示す．

例として，“ kitten ”を“ sitting ”に変形する場合を考える．その流れを以下に示す．

```
kitten
sitten  (“k”を“s”に置換)
sittin  (“e”を“i”に置換)
sitting (“g”を挿入して終了)
```

挿入・削除・置換のそれぞれのコストを 1 に設定した場合，最低でも 3 回の手順が必要とされるので、2 単語間の編集距離は 3 となる．今回の例では，挿入・削除・置換のそれぞれのコストを 1 に設定したが，これらのコストは別々の値を割り振ることも可能である．本手法においては置換コストを 1，挿入・削除コストを 2 に設定している．

3 評価実験

本研究で提案した手法を実装して，実験を行った．今回，amazon.co.jp や価格.com などの様々な Web ページを対象として，抽出対象のテキストと類似する情報を抽出した際の抽出精度の結果を表 1 に示す．表 1 の 1 列目は対象とした Web ページを示していて，＜＞はそのサイトにおける検索キーワードを示している．

表 1 提案手法の適用結果

Webページ	抽出対象	全抽出項目数	抽出した抽出項目数	抽出精度
amazon.co.jp ＜デジタルカメラ＞	商品名	15	15	100%
価格.com ＜ノートパソコン＞	商品名	40	40	100%
価格.com ＜デジタルカメラ＞	価格	60	60	100%
価格.com ＜ノートパソコン＞	評価	20	42	48%
食べログ ＜ラーメン＞	店名	20	20	100%
Yahoo!ショッピング ＜長財布＞	商品名	20	22	90%

ほとんどのサイトにおいて，全抽出項目を高い精度で抽出することができた．また，抽出項目数が増加しても抽出精度が落ちることなく抽出が可能であった．

しかし，価格.com において評価を抽出対象とした場合，抽出精度が低い結果となった．これは，評価の情報と異なる情報であっても同じような HTML パスでソース上で表現されているために，選択した情報と類似していない情報まで抽出してしまったと考えられる．Yahoo!!ショッピングにおいても同様な原因で抽出精度が他のサイトよりも少し低い結果を得た．

次に，編集距離の閾値によって抽出精度がどのように変化するかを検証する．編集距離による抽出精度の変化を図 2 に示す．

今回の実験の場合，編集距離の閾値が“ 3 ”の際に一番高い精度を示していることが明らかになった．閾値が小さすぎても類似する情報をすべて抽出することができず，閾値が大きすぎても選択したテキストと類似しない情報まで抽出する結果となった．

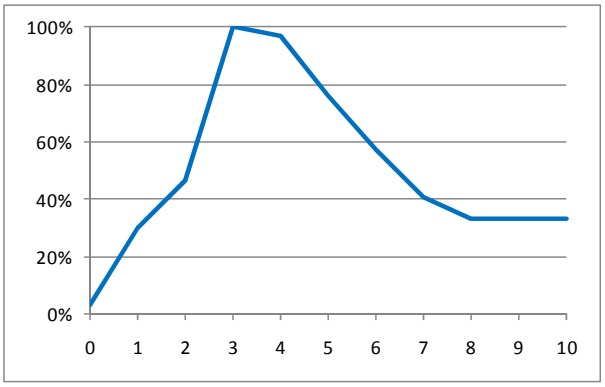

図 2 編集距離と抽出精度の関係

4 結論と今後の展望

本研究では，HTML などの半構造化文書でつくられた Web ページから必要な情報のみを取り出すために，拡張ブラウザを用い，編集距離を組み込んだ Wrapper を生成して Web ページの構造の類似性を求めることで，情報抽出する手法を提案した．提案手法によって Web ページ上からユーザが選択した情報と類似した情報を高い精度で抽出することが可能となった．

しかし，Web ページ上でレイアウトが類似している情報が存在する場合，つまり異なる情報であっても HTML パスが類似している場合，ユーザが選択した情報と異なる情報を抽出してしまう場合があることからレイアウトが類似している場合においても必要な情報のみを抽出できるような Wrappr の生成が今後の課題となる．

参考文献

[1] N. Kushmerick: Wrapper Induction: Efficiency and Expressiveness, *Artificial Intelligence*, Vol.118, pp.15-68, 2000.

[2] C.-H. Chang and S.-C. Lui: IEPAD: Information Extraction Based on Pattern Discovery, the Tenth International Conference of World Wide Web (WWW2001), pp.4-15, 2001.

[3] Utku Irmak, Torsten Suel:Interactive Wrapper Generation with Minimal User Effort, WWW 2006, May 23-26, (2006)

[4] Greasemonkey: http://addons.mozilla.org/ja/firefox/addon/748