DBSJ Journal, Vol.8, No.1
June 2009

教師情報を必要としないニュースページ群からのコンテンツ自動抽出

Primary Content Extraction from News Pages without Training Data

吉田 光男 †　　山本 幹雄 ††

Mitsuo YOSHIDA　　Mikio YAMAMOTO

近年の CMS の普及により Web ページにメニューや広告、著作権表示などが過剰に付加され、ページに占めるコンテンツ（主要部分）は縮小している。本論文では、事前に教師情報を準備する必要のない単純なアルゴリズムで Web ページ群からコンテンツを抽出する手法を提案する。本手法は、Web ページをブロック（コンテンツ及び不要部分の最小単位）の集合であると考え、ある特定のページにのみ出現するブロックはコンテンツであるという単純なアイデアが基になっている。また、本手法のアルゴリズムを実装したソフトウェアを用いて実験を行い、Web 上に存在する日英のニュースページに対して高いコンテンツ抽出性能があることを示す。

In recent years, the proportion of primary content in a Web page has been decreasing as content management systems (CMS's) continue to spread, because CMS's automatically and excessively add unnecessary parts such as menus, advertisements and copyright notices into the Web page. In this paper, we propose a simple method extracting the primary content from a collection of Web pages without training data. We regard a Web page as a set of blocks (minimum unit of primary or non-primary content), and assume that blocks of the primary content are unique and there are copies of those of non-primary content. We describe experimental results to show good performance of the proposed method using real Web pages of the news sites in Japanese and English.

1. はじめに

インターネットが普及した今日、様々な利用者が Web ページを作成し、インターネット上には大量の情報があふれている。2008 年 7 月末に発表された Google のデータによると、Web ページ数は、1998 年から 2008 年にかけて 2600 万ページから 1 兆ページまで急増している [1]。近年の Web ページの増加は、ニュースサイトをはじめとする CMS（Content Management System）[1]の普及に一因がある。CMS は設定したページテンプレートに基づき Web ページを生成するため、誰でも簡単に大量のページを作成することができるようになった。しかし、各 Web ページにメニューや広告、著作権表示が過剰に付加されるようになり、ページに占めるコンテンツ（主要部分）は縮小している。たとえば、図 1 の Web ページ[2]では、ヘッダ、メニュー、広告、関連記事リストなどコンテンツ以外の不要部分が多々存在することにより、ページに占めるコンテンツ（破線部分）の割合が低いことがわかる。Web ページのコンテンツを抽出することができれば、Web 検索システム、携帯電話向けの Web ページ変換システム、コンテンツフィルタリングシステムなどの精度向上及び Web ページを利用する研究促進が期待できる。

図 1　Web ページに占めるコンテンツの例（破線部分）
Fig. 1　A Web page and its content (dashed-line box).

本論文では、事前に教師情報を準備する必要のない単純なアルゴリズムで Web ページ群からコンテンツを抽出する手法を提案する。本手法は、Web ページをブロック（コンテンツ及び不要部分の最小単位）の集合であると考え、ある特定のページにのみ出現するブロックはコンテンツであるという単純なアイデアが基になっている。

以降、2 章では関連研究について、3 章でコンテンツとブロックの定義及び抽出手法について、4 章で評価指標、日米のニュースサイトを対象とした実験結果及び考察について、最後の 5 章でまとめと今後の課題について述べる。

† 学生会員　筑波大学大学院 システム情報工学研究科
m.yoshida@mibel.cs.tsukuba.ac.jp
†† 筑波大学大学院 システム情報工学研究科
myama@cs.tsukuba.ac.jp

[1] Web ページのコンテンツを総合的に管理するシステム
[2] http://www.asahi.com/business/update/0106/TKY200901060314.html

2. 関連研究

Web ページからコンテンツを抽出する手法としては、まず、人手によって抽出ルールを記述する方法が考えられる。この方法の代表例として、正規表現 [2] やラッピング言語 [3] が挙げられる。しかし、インターネット上には無数の Web ページが存在しており、各 Web ページに適した抽出ルールを定めることは、Web サイトごとに定めるにしても、大きな労力を必要とする。

抽出ルールを記述する労力を軽減するために、機械学習と呼ばれる、事前に教師情報を準備しコンテンツ抽出モデルを自動獲得する手法が提案されている。鶴田ら [4] は、平均的な Web ページにおいてウィンドウのどの位置にコンテンツが出現するかを学習する手法を提案している。また、Bing ら [5] は、平均的な Web ページにおいて HTML のどの位置にコンテンツが出現するかを学習する手法を提案している。これらの手法では、事前に教師情報を準備する必要があるため、その準備に労力を必要とする。また、平均的な Web ページの構造が変わると抽出が困難になるという問題を抱えている。ここまでに述べた各手法は、入力として単一の Web ページを受け付ける。

一方、同じ Web サイトで構成された Web ページ群を入力とすることで、教師情報を必要としないコンテンツ抽出手法も提案されている。Lin ら [6] は、サイト内の Web ページを収集し、ページ中の部分の情報量を計算することによりコンテンツを抽出する手法を提案している。この手法では、計算量が大きくなる傾向があるため、Debnath ら [7] は、計算量を小さくした IBDF（Inverse Block Document Frequency）と呼ばれるサイト内におけるページ中の部分の重要度スコアを計算することによりコンテンツを抽出する手法を提案している。Debnath らの手法には大きく分けて 2 つの問題点が存在する。1 つ目の問題点は、コンテンツ候補の抽出に tag-set と呼ばれるコンテンツと不要部分を分断しやすいタグのリストが必要であり、このリストは Web ページデザインの流行に左右されることである。Debnath らは、ニュースページはテーブルタグ（TABLE）によってコンテンツと不要部分が分断される傾向があるため、優先的に分割するのがよいと主張している。しかし、我々の調査では、現在、ニュースページはテーブルタグによってコンテンツと不要部分が分断されておらず、この知識が古くなっていることがわかっている。2 つ目の問題点は、IBDF を計算した後、各 Web サイトに適した閾値を決定しコンテンツを抽出するということである。Web 上には無数の Web サイトが存在しており、全ての Web サイトに適切な閾値を決定することは困難である。

本論文では、ブロックの抽出に W3C（World Wide Web Consortium）が定義するブロックレベル要素を利用することにより、新たにタグのリストを準備する必要のないコンテンツ候補（ブロック）抽出手法を提案する。そして、ある特定の Web ページにのみ出現するブロックはコンテンツであるという単純なアイデアを基に、各 Web ページに閾値を設定する必要がないコンテンツ抽出手法を提案する。

3. Web ページ群のコンテンツ抽出

3.1 コンテンツとは

一般的に、Web ページはユーザが必要とするコンテンツ（主要部分）と、必要としない不要部分から成り立っている。ニュースページを例に取れば、記事タイトルや記事本文はコンテンツであり、メニューや広告は不要部分である。本論文では、ニュースページの記事本文をコンテンツとする。また、記事本文に付随する記事タイトル、記事日時、著者名、写真・図、写真・図の説明文、ニュース配信元の著作権情報もコンテンツとみなす。

不要部分の例としては、広告、メニュー、著作権情報が挙げられる。広告は、表示されている Web ページとの関連性が高ければコンテンツになりうるが、広告除去ソフトウェア[3]が存在するなど一般的にコンテンツと認知されていない。また、メニューは、別の Web ページに移動するための情報であり、その表示されているページに必ずしも必要とされていない。そして、著作権情報は、表示されている Web ページが属する Web サイトの情報が記載されており、メニュー同様、そのページには必ずしも必要とされていない。ただし、ニュース配信元の著作権情報は、表示されている Web ページのコンテンツそのものの権利情報を表しているため、著者名と同列に扱い、コンテンツとして認めている。

3.2 提案手法の概要

我々は、ある特定の Web ページにのみ出現する部分はコンテンツである傾向が高いことに着目した。そして、不要部分は複数の Web ページをまたいで何度も出現するが、コンテンツは 1 つの Web ページにのみ出現する傾向があることがわかった。その結果、何度も出現する不要部分を除外し、他の Web ページには出現しない部分を抽出すればコンテンツを抽出できると考えた。

本論文で提案する Web ページ群のコンテンツ自動抽出手法は、Web ページ群の収集、ブロックの抽出、ブロックのベクトル化、ブロック間の比較、コンテンツの特定の 5 つの過程から構成される。以降の小節で、上の詳細を順次説明する。

3.2.1 Web ページ群の収集

Web ページ群を対象とする従来手法 [6][7] では、同じ Web サイトで構成された Web ページ群を対象としていたが、本手法ではサイトをまたいで構成していても構わない。本手法は、Web ページ収集プログラム（クローラ）とは完全に独立したコンテンツ抽出手法である。ただし、Web ページ群の中に同 Web サイトから収集したページが 2 ページ以上あることが望ましい。

本論文では、Web ページ群を $\boldsymbol{S}$ として次のように表現する。

$$\boldsymbol{S} = \{\boldsymbol{D}_1, \boldsymbol{D}_2, \boldsymbol{D}_3, \ldots, \boldsymbol{D}_N\}$$

$\boldsymbol{D}_i (1 \leq i \leq N, 2 \leq N)$ は各 Web ページとする。

3.2.2 ブロックの抽出

コンテンツを抽出するためには、コンテンツの最小単位を決定する必要がある。本論文では、コンテンツ及び不要部分の最小単位をブロックと呼ぶ。本節では、DOM ツリーからブロックを抽出する手法を述べる。

[3] Adblock (Firefox Add-ons) など

Web ページのレイアウト方法の特徴及び流行を考慮せず、汎用的にコンテンツの抽出を行うためには、ブロックも汎用的に抽出する必要がある。Web ページで利用されている HTML タグは、WWW で使われる技術の標準化を進める国際団体である W3C によって定められており、この定義に従うことで汎用的な抽出が可能になる。W3C の定めた HTML タグは、Web ページ内の見出しや段落など文書の基本構造を構成するためのブロックレベル要素（H1, P, DIV, TABLE など）と、特定の語の修飾やハイパーリンクを設置するためのインライン要素（FONT, STRONG, A など）に大分することができる [8]。

本手法では、ブロックの抽出にブロックレベル要素を用いる。ブロックレベル要素を用いてブロックを抽出する際、ブロックがコンテンツ及び不要部分の最小単位となるよう下位ノードにブロック要素が存在しないように抽出する。ただし、ブラウザにレンダリングされない SCRIPT, STYLE の 2 タグ及びその下位ノードはブロック内に含めない。また、BODY タグはブロックレベル要素ではないが、直下にブロックレベル要素以外が存在する HTML 構造にも対応するため、例外的にブロックとして認める。たとえば、図 2 の DOM ツリー（属性は省略）からブロックを抽出すると、5 つのブロックが抽出される（破線枠部分）。

本論文では、Web ページ $\boldsymbol{D}_i$ を次のように表現する。

$$\boldsymbol{D}_i = \{\boldsymbol{B}_{i1}, \boldsymbol{B}_{i2}, \boldsymbol{B}_{i3}, \ldots, \boldsymbol{B}_{iM_i}\}$$

$\boldsymbol{B}_{ij}(1 \leq j \leq M_i)$ は各ブロックとする。なお、ブロック数 M_i は、Web ページごとに変化するが有限である。

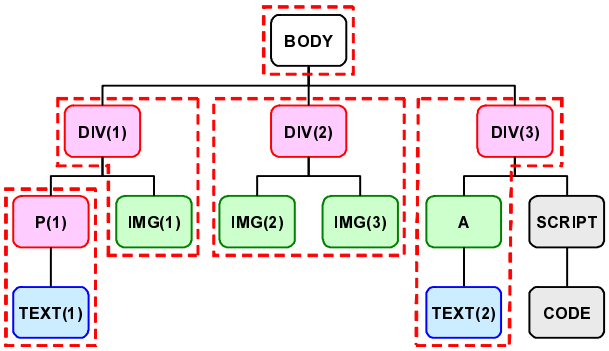

図 2　DOM ツリーから 5 つのブロックを抽出した例
Fig. 2　Five *Blocks* extraction (dashed-line box) from DOM tree.

3.2.3　ブロックのベクトル化

ブロック間の微小な違いを無視して一致を判断するために、各ブロックをベクトルで表現する。本手法では、ブロックのベクトル素性としてブロック内の各タグ、各テキスト、属性 title, alt の各テキストを用いる。そして、各素性の値はそれぞれの出現回数とする。ブロック内の各タグの数は、各ブロックのレイアウト情報を表現している。また、ブロック内の各テキストの数は各ブロックの内容を表現しており、属性 title, alt の各テキストの数は画像（IMG）が出現するブロックの内容を表現している。なお、各テキストをカウントする際、テキストを改行によって分割した結果を小文字に正規化して利用し、空白のみのテキストは除外している。

本論文では、ブロック $\boldsymbol{B}_{ij}$ を次のようにベクトルとして表現し、ブロックベクトルと呼ぶ。

$$\boldsymbol{B}_{ij} = (b_{ij1} \;\; b_{ij2} \;\; b_{ij3} \;\; \ldots \;\; b_{ijL})$$

$b_{ijk}(1 \leq k \leq L)$ はベクトルの各素性の値とする。抽出を行う Web ページ群に含まれる Web ページの数は N であり、テキストはその内容ごとに次元が異なるため L は非常に大きな値を取るが、Web ページ群 $\boldsymbol{S}$ を決定した段階で固定化される。

3.2.4　ブロック間の比較

前節で述べたブロックベクトルを用いて、各ブロックが他の Web ページに出現するかどうか、各ブロック同士を比較する。ブロック同士を比較する際は、ブロックベクトル同士のコサイン類似度を計算する。ブロックベクトル $\boldsymbol{B}_{ij}$ と $\boldsymbol{B}_{kl}$ の類似度 $Sim(\boldsymbol{B}_{ij}, \boldsymbol{B}_{kl})$ は、次のように計算できる。

$$Sim(\boldsymbol{B}_{ij}, \boldsymbol{B}_{kl}) = \frac{\boldsymbol{B}_{ij} \cdot \boldsymbol{B}_{kl}}{\|\boldsymbol{B}_{ij}\|\|\boldsymbol{B}_{kl}\|} \quad (i \neq k)$$

3.2.5　コンテンツの特定

ブロックの一致判断を行ない、他の Web ページには出現しないブロック、すなわち Web ページ群の中で 1 度だけ出現するブロックをコンテンツの一部として抽出する。本手法では、ブロック間の類似度 $Sim(\boldsymbol{B}_{ij}, \boldsymbol{B}_{kl})$ が 0.9 を越えた時、それらのブロックは一致したと認める。コサイン類似度を用いることにより、レンダリングにほとんど影響を与えない微小な違いを無視することができる。なお、この閾値が性能に影響を与えないことは、実験により示す（4.4 節）。

4.　実験と考察

4.1　評価指標

本実験の評価指標は、人手で作成した各データセットの適合率（Precision）、再現率（Recall）、F 値（F-measure）を利用したほか、完全一致率（Perfect-matching）というコンテンツを過不足無く認識できた Web ページの割合も利用した。

本手法のアルゴリズムによって抽出されたブロックの数を N、抽出されたコンテンツのうち正解データと適合していたブロックの数を R、正解データに含まれるブロックの数を C とすると、適合率、再現率、F 値は次のように計算できる。

$$Precision = \frac{R}{N}$$
$$Recall = \frac{R}{C}$$
$$F = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$

完全一致率は、一部の特定コンテンツが各 Web ページで抽出できない場合に低い値を示す。また、ブロック数よりも少ないページ数に依存するため、適合率、再現率、F 値よりも抽出性能に対して敏感に反応する。Web ページ群に含まれる Web ページの数を N、コンテンツを過不足無く認識できた Web ページの数を M とすると、完全一致率は $\frac{M}{N}$ で計算できる。

4.2 日本語のニュースサイトを対象とした実験結果

使用したデータセットは、asahi.com[4]、毎日 jp[5]、YOMIURI ONLINE[6] から収集した計 535 記事である（表 1「データセット詳細」）。各記事の URL は CEEK.JP NEWS[7]から取得し、その URL のリストを基に HTML ファイルを取得した。CEEK.JP NEWS から URL を取得する際は、Web ページの内容にばらつきが出ないよう、政治、経済、スポーツのニュースページのみに絞っている。なお、表中の jpNewsAll は収集した 3 サイトを混合した Web ページ群である。

実験結果を表 1「コンテンツ抽出性能」に示す。図 3 はコンテンツ自動抽出を行った Web ページ[8]の例である（着色部分がコンテンツを示す）。実験結果より、本手法は全体的に高い性能を示していることがわかる。また、表 1 の「マイクロ平均」と「jpNewsAll」の性能がほぼ同等であることから、Web サイトを横断して Web ページ群を作りコンテンツを抽出したとしても、性能にほとんど影響を与えていないことがわかる。一方、毎日 jp の再現率が低く、それに伴い F 値も低い性能を示している。また、完全一致率も比較的低い性能を示している。

再現率が低くなる原因は、重複した Web ページの存在であった。たとえば、図 4 の 2 つの Web ページ[9]は、別の URL であり右上の不要部分（破線部分）も異なるが、コンテンツは同じである。このような例が毎日 jp データセット内に 18 組存在しており、再現率低下の原因となっている。これを解決するには、ブロック間の比較を行う前に Web ページ間の類似度を計算し、類似性の高い Web ページ間ではブロック間の比較を行わないという方法が考えられる。なお、重複した 18 組の Web ページを人手で除外して実験を行ったところ、毎日 jp データセットの抽出性能は、適合率 94.94%、再現率 98.05% を示した。重複した Web ページを検知することにより大幅な性能改善が期待できることがわかる。

完全一致率が低くなる原因は、バリエーションの少ないコンテンツの存在であった。たとえば、図 5 は記事日時の日付（破線部分）の抽出に失敗した Web ページ[10]である。記事日時に時刻情報が含まれない場合、日付のバリエーションが限られるため、他の Web ページにも出現する可能性が高くなる。これ解決するためには、あらかじめ日付フォーマットを学習したモデルが必要になると考えられる。ただし、代表的な日付フォーマットが限られているため、モデル作成は小さな労力で作成できるものと考えられる [9]。

図 3　実験結果（日本語）の Web ページ例（コンテンツ抽出後）
Fig. 3　A Web page (Japanese) and the extracted content (box).

図 4　重複した 2 つの Web ページ例
Fig. 4　Two Web pages that same content.

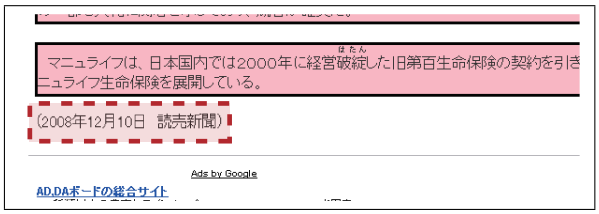

図 5　日付の抽出に失敗した例
Fig. 5　Failed in the extraction at the date (dashed-line box).

[4] http://www.asahi.com/
[5] http://mainichi.jp/
[6] http://www.yomiuri.co.jp/
[7] http://news.ceek.jp/
[8] http://www.yomiuri.co.jp/politics/news/20081205-OYT1T00914.htm
[9] http://mainichi.jp/enta/sports/news/20081211k0000e050032000c.html
http://mainichi.jp/enta/sports/baseball/news/20081211k0000e050032000c.html
[10] http://www.yomiuri.co.jp/atmoney/mnews/20081210-OYT8T00266.htm

表 1　データセットと実験結果（日本語）
Table 1　Data sets and Experimental results (Japanese).

	データセット詳細				コンテンツ抽出性能（%）			
サイト名	記事数	総ブロック数	正解ブロック数	取得日	適合率	再現率	F 値	完全一致率
asahi.com	179	13593	1031	2008-12-12	99.80	97.77	98.78	89.39
毎日 jp	180	28656	1017	2008-12-12	93.72	79.25	85.88	51.11
YOMIURI ONLINE	176	33420	1178	2008-12-12	99.65	95.59	97.57	81.25
マイクロ平均	-	-	-	-	98.00	91.13	94.44	73.83
jpNewsAll	535	75669	3226	-	98.03	91.13	94.46	73.83

4.3　英語のニュースサイトを対象とした実験結果

使用したデータセットは、CNN.com[11]から収集した計 175 記事である（表 2「データセット詳細」）。各記事の URL は Google News（英語版）[12]から取得し、その URL のリストを基に HTML ファイルを取得した。Google News から URL を取得する際は、ドメインのみを指定し[13]、Web ページの内容にばらつきが出るようにしている。ただし、閲覧者がコメントを付けられる Blog 形式のページは人手により除外している。

実験結果を表 2「コンテンツ抽出性能」に示す。図 6 はコンテンツ自動抽出を行った Web ページ[14]の例である（着色部分がコンテンツを示す）。実験結果より、日本語のニュースサイトに比べて比較的低い性能を示している。特に再現率と完全一致率が低い性能を示している。

CNN.com のデータセットには、毎日 jp データセットと同様に、別 URL であるがコンテンツが同じという Web ページが 14 組存在おり、これが再現率が低くなる主な原因であると考えられる。なお、重複した 14 組の Web ページを人手で除外し、実験を行ったところ、適合率 94.11%、再現率 89.53% を示した。日本語のデータセットの場合と同様、重複した Web ページを検知することにより大幅な性能改善が期待できることがわかる。

完全一致率が低くなる原因は、日本語のニュースサイトと同様、バリエーションの少ないコンテンツの存在であった。ただし、日本語のニュースサイトと異なり、著者名とニュース配信元の著作権情報の抽出に失敗しているケースが多かった。日本語のニュースサイトでは、著者名やニュース配信元の著作権情報が記事本文外に明記されない傾向があり、このような違いが発生した。図 7 の Web ページ[15]は、ニュース配信元の著作権情報（破線部分）が抽出できなかった例である。

図 6　実験結果（英語）の Web ページ例（コンテンツ抽出後）
Fig. 6　A Web page (English) and the extracted content (box).

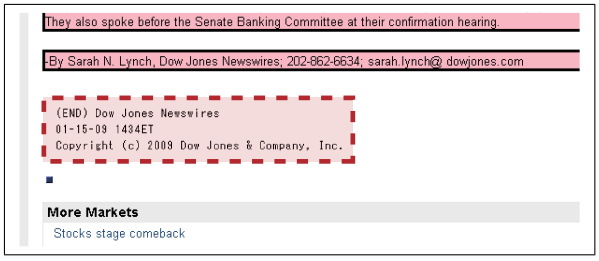

図 7　ニュース配信元の著作権情報の抽出に失敗した例
Fig. 7　Failed in the extraction at the detail of news source (dashed-line box).

[11] http://www.cnn.com/
[12] http://news.google.com/
[13] 検索クエリ「site:cnn.com」を利用した
[14] http://sportsillustrated.cnn.com/2009/baseball/mlb/01/15/bp.salarycap/
[15] http://money.cnn.com/news/newsfeeds/articles/djf500/200901151434DOWJONESDJONLINE001004_FORTUNE5.htm

表 2　データセットと実験結果（英語）
Table 2　Data sets and Experimental result (English).

	データセット詳細				コンテンツ抽出性能（%）			
サイト名	記事数	総ブロック数	正解ブロック数	取得日	適合率	再現率	F 値	完全一致率
CNN.com	175	31401	2758	2009-01-16	94.38	71.28	81.22	29.71

4.4　ブロック間の比較に用いる閾値

本手法では、3.2.4 節で述べた通りブロック間の一致を判定するためにコサイン類似度を用いている。そして、類似度が 0.9 を超えた時、それらのブロックは同じであると認めている。この閾値と抽出性能の関係を示したグラフが図 8 である（jpNewsAll の Web ページ群による抽出実験）。このグラフより、4.1 節で述べた通り完全一致率は F 値よりも敏感に反応しているものの、閾値が 0.75 を超えると抽出性能が安定しており、本手法が閾値に依存しないことがわかる。

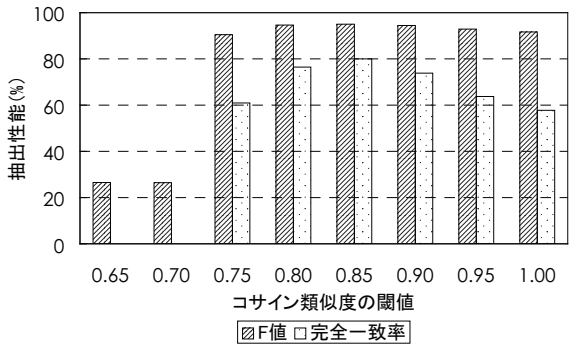

図 8　コサイン類似度の閾値と抽出性能（jpNewsAll）
Fig. 8　Threshold of cosine-similarity and extraction performance (jpNewsAll).

5.　おわりに

本論文では、事前に教師情報を準備する必要のない単純なアルゴリズムで Web ページ群からコンテンツを抽出する手法を提案した。そして、日英のニュースサイトを対象とした実験により、本手法が高いコンテンツ抽出性能があることを示した。本手法は、教師情報や閾値を決定するためのデータを必要としないため、非常に小さな労力で Web ページのコンテンツを抽出することができる。一方、再現率は適合率に比べ低い値を示した。再現率の低下は、Web ページ群の中で行われるコンテンツの再利用、日付などバリエーションの少ない情報の抽出失敗が原因だと考えられ、さらなる改良が必要である。

今後、本手法の抽出結果を利用することにより、単一の Web ページにも適用可能な手法の検討を行う。また、本研究の成果は、コンテンツ自動抽出ソフトウェアという形で公開し、Web ページを利用する研究の標準的なソフトウェアとなることを目指す。

[文献]

[1] Jesse Alpert and Nissan Hajaj. “ We knew the web was big... ”. *Official Google Blog*. 2008-07-25. http://googleblog.blogspot.com/2008/07/we-knew-web-was-big.html, (cited 2009-05-20).

[2] Kevin Hemenway and Tra Calishain. “ Spidering Hacks ”. 村上雅章訳. 初版, オライリー・ジャパン, 2004, 516p., ISBN4-87311-187-0.

[3] 澤菜津美, 森嶋厚行, 杉本重雄, 北川博之. “ HTML ラッパ自動構築手法の提案 ”. 日本データベース学会論文誌. 2008, vol.7, no.1, pp.263-268.

[4] 鶴田雅信, 増山繁. “ 未知のサイトに含まれる Web ページからの主要部分抽出手法 ”. 言語処理学会第 14 回年次大会発表論文集. 東京, March 18-20, 2008. pp.197-200.

[5] Lidong Bing, Yexin Wang, Yan Zhang and Hui Wang. “ Primary Content Extraction with Mountain Model ”. In *Proceedings of IEEE CIT 2008*. Sydney, Australia, July 8-11, 2008. pp.479-484.

[6] Shian-Hua Lin and Jan-Ming Ho. “ Discovering Informative Content Blocks from Web Documents ”. In *Proceedings of ACM SIGKDD 2002*. Alberta, Canada, July 23-26, 2002. pp.588-593.

[7] Sandip Debnath, Prasenjit Mitra, Nirmal Pal and C. Lee Giles. “ Automatic Identification of Informative Sections of Web Pages ”. *IEEE Transactions on Knowledge and Data Engineering*. 2002, vol.17, no.9, pp.1233-1246.

[8] Dave Raggett, Arnaud Le Hors and Ian Jacobs. “ The global structure of an HTML document ”. *HTML 4.01 Specification*. 1999-12-24. http://www.w3.org/TR/1999/REC-html401-19991224/struct/global.html#h-7.5.3, (cited 2009-05-20).

[9] 南野朋之, 鈴木泰裕, 藤木稔明, 奥村学. “ blog の自動収集と監視 ”. 人工知能学会論文誌. 2004, vol.19, no.6, pp.511-520.

吉田 光男　Mitsuo YOSHIDA
筑波大学大学院システム情報工学研究科博士前期課程在学中。情報処理学会、人工知能学会、日本データベース学会各学生会員。
山本 幹雄　Mikio YAMAMOTO
筑波大学大学院システム情報工学研究科教授。自然言語処理の研究に従事。情報処理学会、言語処理学会、ACL 等各会員。