教師情報を必要としない Web ページ群の主要コンテンツ自動抽出

Primary Content Extraction from Blog Pages without Training Data

吉田 光男　　山本 幹雄
Mitsuo YOSHIDA　　Mikio YAMAMOTO

筑波大学大学院システム情報工学研究科
Graduate School of Systems and Information Engineering, University of Tsukuba

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 advertisements, menus and copyright notices into the Web page. We proposed a simple and training data-less method extracting the primary content from a collection of Web pages. 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. In this paper, we show the method is applicable to the primary content extraction from Japanese blog sites.

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 のデータによれば、1998 年には 2600 万ページであった Web ページ数は、現在、1 兆ページまで急増している [Alpert, Hajaj 2008]。近年の Web ページの増加は、ブログ（Weblog）をはじめとする CMS（Content Management System）*1の普及に一因がある。我が国では、2004 年から 2005 年ごろにかけてブログ及びその記事が急増しており、現在も増加傾向が見られる [IICP 2008]。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。すなわち、簡単に大量の情報を発信できるようになった。しかし、各 Web ページにメニューや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は縮小している。たとえば、図 1*2の Web ページでは、ヘッダ、メニュー、広告、関連記事リストなど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、また、Web ページを利用する研究を促すことが期待できる。

我々[吉田, 山本 2009] は、ある Web ページに出現したコンテンツは他のページに出現しないという仮説を立て、国内外のニュースサイトにはその仮説が適用可能であることを示した。しかし、CMS によって生成される代表的な Web ページであるブログの記事ページに対する適用実験が行われていない。本論文では、我々の提案手法がブログの記事ページに対しても適用可能であることを示すとともに、ブログ特有のコンテンツ種別に対する性能評価を行う。

連絡先: 吉田 光男，筑波大学大学院 システム情報工学研究科 知能情報生体工学研究室，〒 305-8573 茨城県つくば市天王台 1-1-1，m.yoshida@mibel.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.100shiki.com/archives/2009/04/lovelogger.html

図 1: Web ページに占めるコンテンツの例（破線部分）

2. 関連研究

Web ページからコンテンツを抽出する手法は、近年、多くの提案が行われている。Bing ら [Bing et al. 2008] は、Web ページを一連のセルと見なし、各セルに文字数、句読点数などに応じたスコアを付加した後、そのスコアの大きさを山に見立て、平均的な Web ページではどの山がコンテンツであるかを学習し抽出する手法を提案している。また、鶴田ら [鶴田, 増山 2008] は、平均的な Web ページにおいて、ウィンドウのどの位置に主要 DOM ノードが出現するかという情報を用いて主要 DOM ノードを抽出し、その主要 DOM ノードの中からヒューリスティックスで不要部分を除去することによりコンテンツを特定する手法を提案している。これらの手法では、事前に教師データを準備する必要があるほか、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。

Lin ら [Lin, Ho 2002] は、同じサイト内の 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の抽出を試みている。この手法では、計算量が大きくなる傾向があり、Debnath ら [Debnath et al. 2005] 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手法を提案している。しかし、彼らの提案手法には、2 つの問題点がある。1 つ目の問題点は、コンテンツ候補となる部分の抽出に 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』情報が必要であり、この情報は Web ページデザインの流行に左右されることである。彼らは、各ニュースサイトではテーブルタグ（TABLE）により記事本文と不要部分が分断されているため、優先的に分割するのがよいと主張しているが、筆者の調査では、現在、各ニュースサイトではテーブルタグ（TABLE）により記事本文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。

我々[吉田, 山本 2009] は、ある Web ページに出現したコンテンツは他のページに出現しないという仮説を立て、Web ページのコンテンツ及び不要部分の最小単位（ブロック）を適切に決定することができれば、他のページに出現しないブロックを抽出することにより、コンテンツ抽出が可能になると考えた。そして、ブロックレベル要素を基にコンテンツ及び不要部分の最小単位である『ブロック』を抽出し、そのブロックが他のページにも出現するか否かを調べることにより Web ページのコンテンツを抽出する手法を提案した。しかし、CMS によって生成される代表的な Web ページであるブログの記事ページに対する適用実験が行われていない。本論文では、我々の提案手法がブログの記事ページに対しても適用可能であることを示す。

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 ページは人間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。本論文の実験対象としているブログの記事ページでは、コンテンツは、さらに情報発信者によるコンテンツと閲覧者等によるコンテンツに大分することができる。本論文では、情報発信者によるコンテンツを記事タイトル、記事本文、記事投稿日時、著者名、写真・図、写真・図の説明文と定義し、また、閲覧者等によるコンテンツをコメントタイトル、コメント本文、コメント投稿日時、コメント著者名、トラックバック[*3]タイトル、トラックバック本文、トラックバック送信日時、トラックバック元ブログ名と定義した。

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[*4]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。ただし、商品レビュー記事の広告に代表されるような、記事本文内において執筆者の意志によって掲載されたと思われるアフィリエイト広告はコンテンツとして認めている。また、メニューは、別の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 Web 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 Web ページには必ずしも必要とされていない。

*3 ブログ特有のリンク通知システム
*4 Adblock (Firefox Add-ons)

3.2 コンテンツ抽出手法の概要

我々が提案した 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、特徴ベクトルの生成、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。

3.2.1 Web ページ群の収集

本研究では、ある Web ページのコンテンツは他の Web ページに出現しないという特定の構造に依存しない仮説を立て、Web ページの集合を与えさえすれば、抽出ルールや閾値を必要とせずにコンテンツを抽出する手法を検討した。

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。

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

$\boldsymbol{D}_i (1 \leq i \leq N)$ は各 Web ページを指す。

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、コンテンツの最小単位を『ブロック』と呼ぶ。ブロックは、コンテンツの最小単位であるとともに、不要部分の最小単位でもある。

提案手法では、ブロックの抽出にブロックレベル要素 [W3C 1999] 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や不要部分の最小単位となるように、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。

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）。なお、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。

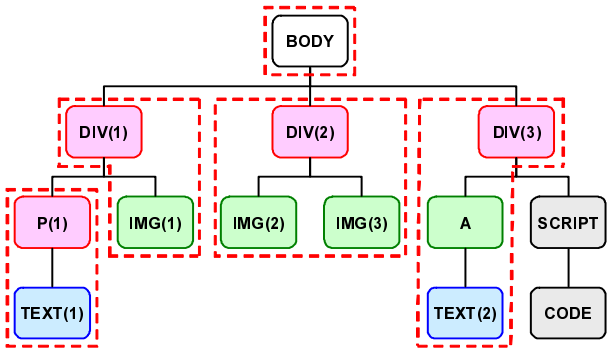

図 2: DOM ツリーからブロックを抽出した例

本論文では、Web ページ群 $\boldsymbol{S}$ に含まれる N 個の Web ページ $\boldsymbol{D}_i (1 \leq i \leq N)$ を次のように表現する。

$$\boldsymbol{D}_i = \{\boldsymbol{B}_{i1}, \boldsymbol{B}_{i2}, \boldsymbol{B}_{i3}, \ldots, \boldsymbol{B}_{iM_i}\} \quad (1 \leq i \leq N)$$

$\boldsymbol{B}_{ij} (1 \leq i \leq N, 1 \leq j \leq M_i)$ は各ブロックを指す。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。

3.2.3 特徴ベクトルの生成

ブロック間の比較を行いコンテンツの抽出を行うため、各ブロックの特徴ベクトルを決定する必要がある。本手法では、各ブロックの特徴ベクトル素性としてブロック内の各タグの数、

各テキスト（小文字に正規化）の数、属性 title, alt の各テキスト（小文字に正規化）の数を用いる。

ブロック内の各タグの数をカウントすることにより、各ブロックのレイアウト情報を表現することができる。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推定することができ、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することができる。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利用し、空白のみのテキストは除外している。

本論文では、Web ページ $\boldsymbol{D}_i(1 \leq i \leq N)$ に含まれるブロックの特徴ベクトル $\boldsymbol{B}_{ij}(1 \leq i \leq N, 1 \leq j \leq M_i)$ を次のように表現する。

$$\boldsymbol{B}_{ij} = (b_{ij1}\ b_{ij2}\ b_{ij3}\ \ldots\ b_{ijL}) \quad (1 \leq i \leq N, 1 \leq j \leq M_i)$$

$b_{ijk}(1 \leq i \leq N, 1 \leq j \leq M_i, 1 \leq k \leq L)$ は特徴ベクトルの各要素を指す。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

3.2.3 節で述べた特徴ベクトルを用いて、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、特徴ベクトル同士のコサイン類似度を計算する。特徴ベクトル $\boldsymbol{B}_{ij}(1 \leq i \leq N, 1 \leq j \leq M_i)$ と $\boldsymbol{B}_{kl}(1 \leq k \leq N, 1 \leq l \leq M_k)$ の類似度 $Sim(\boldsymbol{B}_{ij}, \boldsymbol{B}_{kl})$ は、次のように計算できる。

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

ブロック間の $Sim(\boldsymbol{B}_{ij}, \boldsymbol{B}_{nm})$ が 0.9 を越えた時、それらのブロックは同じであると認める。コサイン尺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若干の違いを吸収することができる。

3.2.5 コンテンツの特定

3.2.4 節で述べたブロック間の比較方法により、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抽出する。

4. 実験と考察

4.1 評価指標

本実験の評価尺度は、2. 節で述べた先行研究に倣い、人手で作成した各データセットに含まれるコンテンツの適合率（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。

4.2 実験結果

4.2.1 コンテンツ抽出

使用したデータセットは、livedoor Reader*5の登録数ランキング上位からブログ形式の 9 サイト、100SHIKI*6、Engadget Japanese*7、ネタフル*8、404 Blog Not Found*9、IDEA*IDEA*10、My Life Between Silicon Valley and Japan*11、Going My Way*12、TechCrunch Japan*13、Life is beautiful*14 から収集した計 206 記事である（表 1「データセット詳細」）。コンテンツの候補となるブロックは 35216 ブロック存在し、人手によってコンテンツと認められるブロックは 3822 ブロックであった。なお、表中の jpBlogAll は収集した 9 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（完全一致率は PM と表記）。実験結果より、提案手法は全体的に高い性能を示していることがわかる。一方、コンテンツを過不足無く認識できた Web ページは非常に少なかった。図 3*15はコンテンツ自動抽出を行った Web ページの例である（着色部分がコンテンツを示す）。上部のナビゲーション部分が誤認識されているが、概ねコンテンツが抽出できている。

図 3: 実験結果の Web ページ例（コンテンツ抽出後）

4.2.2 コンテンツ種別の再現率

3.1 節で述べたとおり、ブログ記事のコンテンツは情報発信者によるコンテンツと閲覧者等によるコンテンツに大分できる。本論文では、どちらのコンテンツに属しているかを自動判別する手法の検討は行っていないため、コンテンツ種別の性能に関しては、再現率のみを用いて評価した。なお、人手によって情報発信者によるコンテンツと認められるブロックは 2927 ブロックであり、閲覧者等によるコンテンツと認められるブロックは 973 であった。

情報発信者によるコンテンツと閲覧者等によるコンテンツの再現率を表 1「コンテンツ種別再現率」に示す（「-」は該当コンテンツが存在しなかったこと示す）。表より、情報発信者によるコンテンツよりも閲覧者等のコンテンツの方が全体的には抽出がやや容易であることがわかるが、サイトによってばらついていることがわかる。

4.3 考察

適合率が悪くなる原因を調べたところ、記事タイトルを含むナビゲーションリンク（図 3 の上部）、トラックバック送信先情

*5 http://reader.livedoor.com/
*6 http://www.100shiki.com/
*7 http://japanese.engadget.com/
*8 http://netafull.net/
*9 http://blog.livedoor.jp/dankogai/
*10 http://www.ideaxidea.com/
*11 http://d.hatena.ne.jp/umedamochio/
*12 http://kengo.preston-net.com/
*13 http://jp.techcrunch.com/
*14 http://satoshi.blogs.com/
*15 http://kengo.preston-net.com/archives/004082.shtml

表 1: データセットと実験結果

	データセット詳細			コンテンツ抽出性能				コンテンツ種別 再現率	
ブログ名	記事数	ブロック	コンテンツ	適合率	再現率	F 値	PM	情報発信者	閲覧者等
jpBlogAll	206	35216	3900	0.9353	0.8421	0.8863	0.0097	0.8288	0.8818
100SHIKI	10	725	165	0.9463	0.8545	0.8981	0.0000	0.8693	0.6667
Engadget Japanese	40	6163	235	0.8333	0.6596	0.7363	0.0500	0.6231	0.8611
ネタフル	30	2494	455	0.8646	0.8703	0.8675	0.0000	0.8692	1.0000
404 Blog Not Found	50	19264	1755	0.9948	0.8781	0.9328	0.0000	0.8567	0.9067
IDEA*IDEA	9	569	127	0.9250	0.8740	0.8988	0.0000	0.8843	0.6667
My Life Between ...	5	575	212	0.9099	0.9528	0.9309	0.0000	1.0000	0.7826
Going My Way	30	1950	521	0.9336	0.8100	0.8674	0.0000	0.8100	-
TechCrunch Japan	20	1951	190	0.8543	0.6789	0.7566	0.0000	0.6758	0.7500
Life is beautiful	12	1525	240	0.8291	0.9500	0.8854	0.0000	0.9147	0.9910

報（送信先 URL など）を誤って抽出しているケースが多かった。そして、これらの内容はどの記事ページにも含まれる傾向があり、完全一致率の大幅低下の原因であると考えられる。

再現率が悪くなる原因を調べたところ、写真のみのコンテンツの抽出に失敗しているケースが多かった。これは、イメージタグ（IMG）に属性 title, alt が適切に記述されておらず、特徴ベクトルに内容を推定する情報が含まれていないためだと考えられる。また、見かけ上は段落になっていても、段落タグ（P）が利用されておらず、本来であれば複数のブロックに分けられるべき部分が 1 つのブロックになっているケースも存在した。この場合、段落に見せるための改行タグ（BR）の影響が強くなり、ブロック間の比較においてテキストの情報を実質的に加味せず、同じブロックであると判定されてしまう。これらの問題を解決するためには、特徴ベクトルの生成方法とブロック間の比較方法を改善する必要がある。提案手法では、レンダリングにほとんど影響を与えない若干の違いを吸収するために、特徴ベクトルをタグとテキストのみにしぼり、コサイン尺度を用いてブロック間の比較を行っているが、属性情報や並び順を含め、完全一致による比較にすることで再現率は改善するものと考えられる。

適合率、再現率の計算においてブロックのサイズが加味されておらず、テキスト 10 文字のブロックが抽出できても、テキスト 1,000 文字のブロックが抽出できても性能評価に与える影響はどちらのブロックも同じである。しかし、コンテンツ抽出結果の利用を考えた場合、テキスト 10 文字のブロックよりもテキスト 1,000 文字のブロックの方が重要であると考えられる。そのため、ブロックのサイズを加味した性能評価方法を検討する必要がある。

5. おわりに

本研究では、ある Web ページに出現したコンテンツは他の Web ページに出現しないという仮説によるコンテンツ抽出がブログ記事にも適用可能であることを示した。この仮説による実装は、一切の教師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。

今後は、本研究の成果をソフトウェアとして公開し、Web ページに関する研究の標準的なソフトウェアとなることを目指す。また、情報発信者によるコンテンツと閲覧者等によるコンテンツの分離手法の研究を行うことを考えている。

参考文献

[Alpert, Hajaj 2008] Jesse Alpert, Nissan Hajaj. (2008). “ We knew the web was big... ”. Official Google Blog. http://googleblog.blogspot.com/2008/07/we-knew-web-was-big.html, (Accessed 2009-04-14).

[Bing et al. 2008] Lidong Bing, Yexin Wang, Yan Zhang, Hui Wang. (2008). “ Primary Content Extraction with Mountain Model ”. IEEE CIT2008. pp.479-484.

[Debnath et al. 2005] Sandip Debnath, Prasenjit Mitra, Nirmal Pal, and C. Lee Giles. (2005). “ Automatic Identification of Informative Sections of Web Pages ”. IEEE Transactions on Knowledge and Data Engineering. Vol.17, No.9, pp.1233-1246.

[IICP 2008] 総務省 情報通信政策研究所（IICP）. (2008). “ ブログの実態に関する調査研究 ”. http://www.soumu.go.jp/iicp/chousakenkyu/data/research/survey/telecom/2009/2009-02.pdf, (Accessed 2009-04-14).

[Lin, Ho 2002] Shian-Hua Lin, Jan-Ming Ho. (2002). “ Discovering Informative Content Blocks from Web Documents ”. In Proceedings of ACM SIGKDD’02. pp.588-593.

[鶴田, 増山 2008] 鶴田 雅信, 増山 繁. (2008). “ 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.

[吉田, 山本 2009] 吉田 光男, 山本 幹雄. (2009). “ 教師情報を必要としない Web ページ群のコンテンツ自動抽出ツールの提案 ”. 第 1 回データ工学と情報マネジメントに関するフォーラム（DEIM2009）.

[W3C 1999] W3C. (1999). “ The global structure of an HTML document ”. HTML 4.01 Specification. http://www.w3.org/TR/1999/REC-html401-19991224/struct/global.html#h-7.5.3, (Accessed 2009-04-14).