情 報 学 基 礎　68－6
自 然 言 語 処 理 151－6
（２００２．９．１８）

HTMLの表形式データの変換と携帯端末表示への応用

塚本 修一† 増田 英孝† 中川 裕志‡

† 東京電機大学工学部 〒101-8457 東京都千代田区神田錦町 2-2
‡ 東京大学情報基盤センター 〒113-0033 東京都文京区本郷 7-3-1
E-mail:shu1@cdl.im.dendai.ac.jp,masuda@im.dendai.ac.jp,nakagawa@dl.itc.u-tokyo.ac.jp

概要

本研究は、HTMLの表形式データの構造の認識とその後の利用を目的とした変換のために、表の項目名と項目データの境界を認識するシステムを実現した。表はデータを整理し、見やすくする性質がある。しかし、携帯端末などの低解像度小画面にHTMLの表を表示する場合、スクロールすると項目名の部分が見えなくなってしまう。また、罫線が引かれている為に、表示領域にも制限が出来、単語途中の折り返しにより可読性が低下する。そこで、本研究では、表のデータをユーザが要求する形に出力するための基礎技術として、HTMLの表の構造を認識するアルゴリズムを提案する。提案手法は、表の行間あるいは列間の類似度による。すなわち類似度が低い場合には、行間あるいは列間に内容的な切れ目があると認識する。このアルゴリズムを実際のWebページ上の表データに適用したところ80%程度の認識率を得た。

Recognition of HTML Tables and Transformation for Displaying on Mobile Terminal Screen

†Shuichi TSUKAMOTO †Hidetaka MASUDA ‡Hiroshi NAKAGAWA
† School of Engineering, Tokyo Denki University
2-2 Kandanishiki-cho, Chiyoda, Tokyo 101-8457, Japan
‡ Information Technology Center,The University of Tokyo
7-3-1 Hongo, Bunkyo, Tokyo 113-0033, Japan
E-mail:shu1@cdl.im.dendai.ac.jp,masuda@im.dendai.ac.jp,nakagawa@dl.itc.u-tokyo.ac.jp

abstract

We implemented a recognition system to identify the boundary between attribute names and values of a table in HTML in order to obtain its structure. Table in HTML is aimed at displaying information clearly and understandably. However, users can't see the attributes of the table by using PDA, because of its small and low resolution display when they browse the Web pages. Its low readability is caused by the phenomena such that only a small portion of table is shown on the screen at once, and original one line is usually broken up into many lines on display screens. We propose an algorithm to recognize the structure of tables in HTML for the purpose of transforming them into forms of high readability even on a small screen of mobile terminal. Our method utilizes a similarity between rows(or columns)of the table. Precisely speaking, if we find an adjacent pair of rows(or columns) having low similarity, they probably are boundaries between item name row(or column)and item data rows(or columns). We achieved approximately 80% accuracy of recognition by applying our algorithm to existing tables on the Web.

1　はじめに

近年、携帯電話や PDA などの携帯端末から Web ページをブラウズしたいという要求が急増している。しかし、現状では、PC の高解像度大画面 (解像度が最低でも 640 × 480 以上) を前提として作られているページがほとんどである。携帯端末デバイスの画面解像度は年々高くなっているが、携帯端末の画面サイズは限られているために読める大きさで表示できる文字数には物理的限界がある。また、画面をスクロールさせるための操作量が増加する。さらに、Web ページ上の表を表示する際に、ブラウザによって <TABLE> タグの取り扱い方や、対応するタグの種類が異なるため、その表示に問題が発生する [1]。これらの問題を解決するためにさまざまな研究が現在なされている [2]。 また、コンテンツを端末に向けて動的に生成する方式も提案されている [3]。現状では、既存のコンテンツを新たに人手によって作り直しており、機械処理で自動的に行われるには至っていない。そこで、本研究では高解像度大画面向けに作られた既存の Web ページを携帯端末でブラウズする際の表の表示に問題点をしぼり、まず、表の項目名、項目名に対応するデータ (以下、「項目データ」と呼ぶ) の境界を同定することにより、その構造を認識するアルゴリズムを提案し、評価した。次に、このアルゴリズムを用いて表を携帯端末に適した形に自動変換して表示するシステムを実装した。以下、2. では、携帯端末における表の表示の問題点を挙げ、3. では、表の構造認識システムとして採用したベクトル空間法によるセル (表の 1 つの升目) の類似度の定義と計算、及びアルゴリズムを提案し、実験的に評価している。4. では、システムを用いて表を変換した結果を示し、5. でまとめを述べる。

2　携帯端末における表示の問題点

携帯電話、PDA などの携帯端末を用いて Web ページをブラウズする際には、小画面、低解像度のためにさまざまな問題が発生する。ここでは、具体的な問題点を挙げその解決方法の提案を行う。

2.1　携帯端末で表を表示する際の問題点

表は、本来情報を整理し分かりやすくするために作られている。しかし、小画面低解像度の携帯端末で表をブラウズすると、逆に可読性が低下し、読み誤りが生じることがある。また使用するブラウザによって表示が異なる場合がある。図 1 に PC で表を含むページを表示した例を示す。解像度が高く画面サイズが大きいため、表全体を見渡すことができる。図 2 に PalmsOS[4] 上の AvantGo[5] ブラウザ、図 3 に Palmscape[6] ブラウザで図 1 と同一の表を含むページを表示した例を示す。

(人)

	総数	30～39歳	40～49歳	50～59歳	60～69歳	70歳以上
総数	8369	1480	1660	1995	1701	1533
男性	3854	682	777	928	832	635
女性	4515	798	883	1067	869	898

図 1: PC 画面での表の表示例

図 2: AvantGo での表示例

図 2 の AvantoGo では、罫線が表示されないために表の行と列の関係を保持することが難しい。次に、図 3 の Palmscape では罫線が表示されているので行と列の関係を認識できるが、小画面低解像度のために以下の問題が発生する。第 1 に、各セルの横幅が狭くなるためにセルデータの途中

図 3: Palmscpae での表示例

図 4: スクロールした時の Palmscape での表示例

で折り返しが発生し、読み誤りを起こす可能性がある。第 2 に図 4 は、図 3 の画面をスクロールしたものであるが、表の項目名の部分が隠れてしまい、表の各セルが何を示すか見失ってしまう。その結果、スクロールしてページを戻さなくてはならない。表の行と列の数が大きくなればなるほどこれら 2 つの問題が顕著となる。また、表の <TD>,<TH> タグの colspan,rowspan オプションの値が増加すると、1 つのセルデータを 1 画面内に収めて表示できなくなり、さらに可読性が低下する。図 5 にその例が顕著に表れたものを示す。

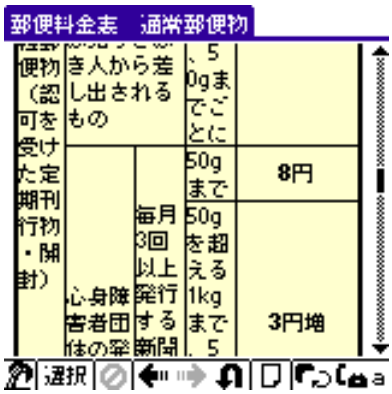

図 5: rowspan オプションで表示した例

2.2 Web ページ上の表の種類及び型

Web コンテンツ中の <TABLE> タグの利用目的は、以下の 3 種類に分類 [7] できる。

レイアウト

<TABLE> タグの BORDER 属性が 0 であり、ページのレイアウトを整えるために使われている。

本質的な表

本質的な表では項目名に対応して、項目データが列挙される構造を持つ。<TABLE> タグの BORDER 属性が 1 以上であり、2 セル以上から構成される。

特殊型

セルが 1 つであり <TABLE> タグの BORDER 属性を 1 以上として、強調表現を行う。

本質的な表は、項目名と項目データから構成される [8]。この項目名と項目データの位置によって Web ページの中の表を 3 つの型に分類する。

時間割型

行、列どちらにも項目名を持っている表である。

縦一覧型

最初の数行が項目名となっており、それ以降の行のデータが項目データとなっている表である。

横一覧型

横一覧型は縦一覧を転置して、最初の数列が項目名となっており、それ以降の列のデータが項目データとなっている表である。

本研究では、これら 3 つの型に対応する構造認識及び、項目名:項目データのペアへの変換システムを実装した [9] ので、次節にて詳しく述べる。

3 表の構造認識システム

3.1 システムの概要

本システムは、2.1 で述べた本質的な表のみを対象する。これまでに、表の研究では言語的性質を点数化し表の表すドメインを認識する研究 [10][11] があるが、複雑な表や、セルの複数に属性があるもの、また未知のドメインには対応できない。

本研究ではセル間の類似度をベクトル空間法によって計算し、類似度の比を用いて、行と列の項目名と項目データを計算し区別する。なお、本システムでは、3 行 3 列以上の大きさを持つ表を認識の対象にしている。

提案するシステムはおおよそ次のような構造である。まず、3.2 に述べるように表の正規化をする。次に、3.3 で述べるように表の各セルをその内容の言語的性質によってベクトル表現する。行と行、あるいは列と列の類似度をベクトルで表現されたセル間の *cosine* を用いて計算する。さらに、この類似度を用いて、行あるいは列における項目名、項目データの結果を求める。これについては、3.4 で詳述する。最後に 3.5 で述べるように、境界判定のため閾値を 10fold 交差検定で学習し、アルゴリズムの精度を実験的に評価する。

3.2 表の正規化

図 6 に示すように表は rowspan、colspan オプションにより 2 つ以上のセルを結合している場合がある。この場合は図 7 の様に正規化する [12]。

		食物名		
		リンゴ	バナナ	ミカン
栄養	カルシウム (mg)	10.1	2.1	3.5
	ビタミン C(mg)	1000	2764.4	349
	亜鉛 (pg)	376.2	3776.3	763.0

図 6: rowspan、colspan オプションがある表

		食物名	食物名	食物名
		リンゴ	バナナ	ミカン
栄養	カルシウム (mg)	10.1	2.1	3.5
栄養	ビタミン C(mg)	1000	2764.4	349
栄養	亜鉛 (pg)	376.2	3776.3	763.0

図 7: rowspan、colspan オプションを外して正規化した表

3.3 ベクトルの要素とベクトルの計算

表の i 行 j 列のセルを $Cell_{ij}$ として、各セルの N 個の言語的性質 $k = 1, \ldots, N$ に対応して、その性質を持てば 1、持たなければ 0 と値 w_k を定義する。w_k を要素とするベクトルを式(1) のように定義する。

$$\overrightarrow{Cell_{ij}} = (w_1, w_2, \cdots, w_N) \tag{1}$$

以下にベクトルの要素となる言語的性質を列挙する。今回の実験では N は合計で 93 であり、内容は以下に示す。

ベクトルの各要素

- 連続データ (2 次元)

  行、列を基準として、“1,2,3,···” 等のある決まった連続性を持ったデータ列や、“リンゴ, ミカン, バナナ,···” 等、“果物” として同一クラスに含まれる各々のデータ群を、1 つのベクトルの次元として定義する。

- 句読点 (1 次元)

  項目名は句読点のない簡潔な文字列で表されていることが多い。よって、句読点がないことを 1 つのベクトルの次元として定義する。

- 文字長 (3 次元)

  項目名は文字長が短いことが多い。文字長が 0(空白)、半角 10 文字以内、半角 11 文字以上をそれぞれベクトルの次元とした。

- 接頭辞 [13](14 次元)

  “第”,“平成”,“特” など 14 種の接頭辞の各々にベクトルの次元を割り当てる。

- 接尾辞 [13] (43 次元)

  “日”,“課”,“年” など 43 種の接尾辞の各々にベクトルの次元を割り当てる。

- 単位 (17 次元)

  “kg”,“人”,“円” など 17 種の単位の各々にベクトルの次元を割り当てる。

- 特殊文字 (11 次元)

  項目名として、一定の期間を表している “〜” や、備考などを示す “ (” ,“) ” などが使われることが多いことから、それら 11 種の各々にベクトルの次元を割り当てる。

- テーブルタグの属性 (2 次元)

  一般的に colspan,rowspan のセル内あるいは、colspan のセルの直下、rowspan のセルの直後の表データは項目名となることが多い。ある表データが、colspan あるいは rowspan の構造中に存在するか、あるいは colspan のセルの直下、rowspan の直後のセルであれば、各々をベクトルの次元に割り当てる。

  テーブルタグは構造の決定に重要な役割を持っているため、1 以上の値をとる場合もある。これにより、colspan あるいは rowspan に関係する表データと、そうでない表データとの距離を離すことができる。

3.4 認識アルゴリズム

ベクトルの計算

m 行 n 列の表の行間、あるいは列間の類似度を計算するために、まず表の i 行 j 列のセルを $Cell_{ij}$ として表し、同じ列の $Cell_{kj}(k \neq i)$ との類似度の平均 $Sim_{row}(i,j)$ を次式で求める。

$$Sim_{row}(i,j) = \frac{1}{m-1}\sum \frac{\overrightarrow{cell_{ij}}\cdot\overrightarrow{cell_{kj}}}{\left|\overrightarrow{cell_{ij}}\right|\left|\overrightarrow{cell_{kj}}\right|} \quad (2)$$

ここで、$\sum$ の範囲は、$k = 1, \cdots, n$、但し、$k = i$ は除く。$\overrightarrow{cell_{ij}}\cdot\overrightarrow{cell_{kj}}$ は、$\overrightarrow{cell_{ij}}$ と $\overrightarrow{cell_{kj}}$ の内積を表し、$\left|\overrightarrow{cell_{ij}}\right|$ と $\left|\overrightarrow{cell_{kj}}\right|$ は、それぞれ $\overrightarrow{cell_{ij}}$ と $\overrightarrow{cell_{kj}}$ の絶対値を表す。したがって、$\sum$ の内側の式は、$\overrightarrow{cell_{ij}}$ と $\overrightarrow{cell_{kj}}$ の $cosine$ である。図 8 で $Cell(1,1)$

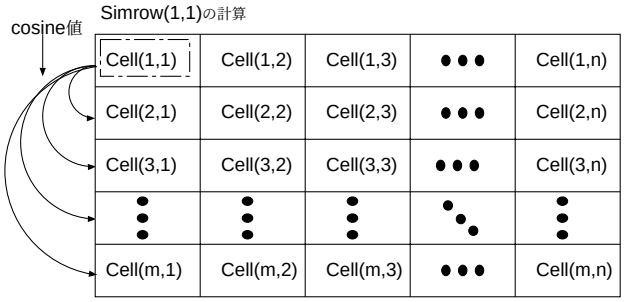

図 8: $Sim_{row}(1,1)$ の計算

と第 1 列中のセルとの類似度の計算の様子を示した。次に、第 i 行のセル、即ち $Cell_{ij}(j = 1, \cdots n)$ のすべてについて $Sim_{row}(i,j)$ を計算し、その行と他の行との類似度の平均 $Sim_{row}(i)$ を次式で求める。

$$Sim_{row}(i) = \frac{1}{n}\sum_{k=1}^{n} Sim_{row(i,k)} \quad (3)$$

式(3) で計算した結果を図 9 に示す。$Sim_{row}(i)$ の値は、第 i 行が、他の行と類似していれば大きく、類似していなければ小さくなる。

項目名を表す行と項目データを表す行とは類似度

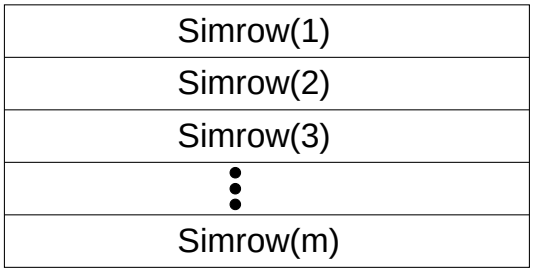

図 9: 式(3) の計算結果

が低い。一方、項目データを表す行同士は類似度が

高い。また、項目名を表す行はWebページでは上にくることが一般的である。$i=1$が第1行である。例えば、1行目と2行目の間が項目名と項目データの境界なら図10のようになる。よって、$Sim_{row}(i)$

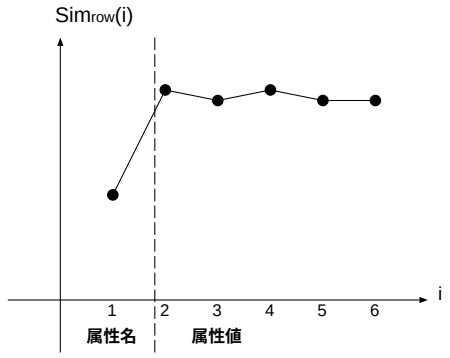

図 10: 項目名と項目データの境界におけるSim_{row}の変化の様子

とi行以下の$Sim_{row}(i+1),\cdots,Sim_{row}(m)$の平均の比$R(i)$を、式(4)のように定義すると、

$$R(i)=\frac{Sim(i)}{\frac{1}{m-i}\sum_{k=i+1}^{m}Sim(k)} \tag{4}$$

1. i行が項目名、$i+1$行以下が項目データなら$R(i)$は小さい
2. i行以下が全て項目データなら$R(i)$は大きい

よって、項目名と項目データの行の境界Tは次のアルゴリズムで求まる。但し、θは、境界かどうかを判定する閾値である。

```
T = 0;
for(i=1;i<=m;i++){
    if(R(i)< θ){ T = i; }
    else{ break; }
}
if(T==0){ 縦方向に境界なし }
else{T 行までが項目名の行 }
```

以上は項目名の行と項目データの行の境界を求めるアルゴリズムだが、以上の導出において、縦横を交換すれば、$Sim_{col}(j)$を計算でき、そして項目名の列と項目データの列の境界を認識できる。以上のアルゴリズムによって、切り出された結果から2.2の表の型に当てはめる。

縦一覧

行の判定でTの値が1以上、列の判定でTの値が0のとき、縦一覧型とする。

横一覧

行の判定でTの値が0、列の判定でTの値が1以上のとき、横一覧型とする。

時間割

行の判定でTの値が1以上、列の判定でTの値が1以上のとき時間割型とする。

3.5 認識アルゴリズムの評価実験

さて、3.4で述べたアルゴリズムで$R(i)$の大きさの判定に用いる閾値θを最適化しなければならない。これは、人手で作った正解によって実験的に決める。そこで、本アルゴリズムの評価には、θの最適化を含め10 fold 交差検定によって評価した。但し、本システムの評価に使用する表の大きさは、3行3列以上の大きさを持つ表を対象とした。

2行n列、m行2列の表については、第1行、第1列がそれぞれ1つの属性を表現している[14]。このため、評価対象から除外する。

まず、最適な閾値θを求めるための教師データとして、Web上にある154の表を人手によって項目名と項目データの行あるいは列の境界を決めた。この教師データによって10 fold 交差検定を行った。その結果、行の閾値θは0.90、列の閾値θは0.70となった。

評価の結果を表1、2に示す。また、評価を行った表の大きさの平均は9行6列であり、それぞれの型の個数とその内訳を表3に示す。

この表3の結果の内、時間割型となるものは66個あり、切れ目の内訳は1行目1列目が43個、2

表 1: 行方向の結果

データの種類	正解率
トレーニングデータ	82.17%
テストデータ	81.88%

表 2: 列行方向の結果

データの種類	正解率
トレーニングデータ	78.11%
テストデータ	77.00%

表 3: 交差検定によるテストデータとして評価をした 154 表の内訳

		切れ目の行 (or 列)			合計
		1	2	3	
型	縦	103	25	1	129
	横	68	2	0	70

行目 1 列目が 22 個、2 行目 2 列目が 1 個である。この結果から、システムはおよそ 80% の正解率で表の項目名を認識することができる。残りの 20% の表は項目名の部分にもかかわらず、言語的類似度がすべて高く認識できない表 (8%)、逆に項目データの部分にもかかわらず、言語的類似度が低い表 (12%) の 2 つに大別できる。

4 表示変換

3.5 で認識した項目名と項目データを携帯端末で理解しやすい形に表示するための変換の方針としては、常に項目名と項目データをペアで表示することにした。これは、2 における考察の結果、項目名と項目データが乖離して読み難くなっていることが分ったので、それを回避するための方策である。これによって、スクロールしても表が示す内容を見失うことがなくなる。表示領域の制限が緩和され、単語途中の折り返しにより可読性が低下することを避けることができる。システムが求めた結果を使って図 1 の表を変換した例を図 11 に示す。はじめに列の項目名の“ 男性 ”を表示し、次にそれらに付随する行の項目名と、そのペアの値を表示してあり、図 2、図 3 よりは理解しやすい。

図 11: 図 1 の表をシステムで変換した例

5 まとめ

本稿では表形式データの変換のために表の項目名と項目データを切り出すシステムについて述べた。提案したアルゴリズムを適用したシステムはおよそ 80%の正解率で項目名と項目データを認識することができる。

今後の課題として、2 行 n 列、m 行 2 列の認識がある。現段階では、経験的に m 行 2 列の表に関しては横一覧である場合が多く [9]、横一覧としている。2 行 n 列 (n は 3 以上) は、縦一覧としている。2 行 n 列、m 行 2 列の表はセル同士の距離が、同じになることは明らかであり、ベクトル空間法では類似度を決定することができない。また、類似度の各々のベクトル要素の値は 1 か 0 としているため、ベクトルの最適化が必要である。ユーザの好みに応じて表のデータの並べ換えを行ったり、表の任意の行や列を選択して表示するユーザインターフェースを実装する予定である。

参考文献

[1] 北山文彦, 広瀬紳一：Dharma さまざまなインターネット端末にコンテンツを適応させるソフトウェア技術, 情報処理学会, Vol. 142, No. 6, pp. 576–581 (2001).

[2] 中川裕志：モバイル端末向けコンテンツ記述,

言語処理学会第 8 回大会併設ワークショップ, pp. 33–41 (2002).

[3] VertexLinkCorporation:C3GATEServer, http://www.vertexlink.co.jp/.

[4] パームコンピューティング株式会社, http://www.palm-japan.com/.

[5] AvantGo,Inc:AvantGo4.2, http://avantgo.com/.

[6] 株式会社イリンクス:Palmscape3.1.1J, http://www.ilinx.co.jp/.

[7] MASUDA, H., YASUTOMI, D. and NAKAGAWA, H.: How to Transform Tables in HTML for Displaying on Mobile Terminals, *6th NLPRS2001 Workshop of Automatic Paraphrasing:Theories and Applications*, pp. 29–36 (2001).

[8] YOSHIDA, M.: Extracting Attributes and Their Valuse from Web Pages, *ACL-02 Student Research Workshop*, pp. 72–77 (2002).

[9] 安富大輔, 増田英孝, 中川裕志：携帯端末によるテーブル認識変換システムの構築と評価, 言語処理学会第 8 回年次大会, pp. 347–350 (2002).

[10] HURST, M. and DUGLAS, S.: Layout and Language: Preliminary Experiments in Assingning Logical Structure to Table Cells, *Proceedings of the Fifth Conference on Applied Natural Language Processing*, pp. 217–220 (1997).

[11] HURST, M. and DUGLAS, S.: Layout and Language:Preliminary investigations in recognizing the structure of tables, *In Proceedings of the Fourth International Conference on Document Analysis and Recognition(ICDAR)*, pp. 1043–1047 (1997).

[12] Chen, H.-H., Tsai, S.-C. and Tsai, J.-H.: MiningTables from Large Scale HTML Texts, *COLING2000*, pp. 166–172 (2000).

[13] 塚本修一, 安富大輔, 増田英孝, 中川裕志：HTML 文書における表の携帯端末のための構造変換, 第 64 回情報処理学会全国大会, pp. 93–94 (2002).

[14] 伊藤史朗, 大谷紀子, 上田隆也, 池田祐治：属性オントロジーの抽出と統合を用いた実空間と情報空間のナビゲーション システム, 人口知能学会, Vol. 14, No. 6, pp. 69–77 (1999).