HTMLの表形式データの構造認識と携帯端末表示への応用

増田英孝[†] 塚本修一[†]
安富大輔[†,☆] 中川裕志[††]

本研究では，HTML の表形式データに対して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．携帯端末などの低解像度小画面上に表を表示する場合，スクロールすると項目名の部分が見えなくなる．また，表示領域が複数のセルに分割されるため，単語途中の折り返しが頻繁に発生し可読性が低下する．そこで，表のセル間の類似度を計算するために言語的性質を利用し，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アルゴリズムを提案する．このアルゴリズムを実際の Web 上の表データに適用したところ，行方向で 82%，列方向で 78%の認識率を得た．認識の結果を用いて，携帯端末向けに表を項目名と項目データの組として表示変換し，表示した結果を被験者を用いて評価した．

Recognition of HTML Table Data and Its Application for Displaying on Mobile Terminal Screen

Hidetaka Masuda,[†] Shuichi Tsukamoto,[†] Daisuke Yasutomi[†,☆] and Hiroshi Nakagawa[††]

We implemented a recognition system to identify the boundary between attribute names and values of a table in HTML. 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. 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 82% accuracy of recognition of lengthways (and 78% accuracy of recognition of sideways) by applying our algorithm to existing tables on the Web. By using the result, we transform the original table into the attribute-value pairs for a small screen of mobile terminal, and evaluate them by human subjects.

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増加している．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，人間が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，1 画面に表示できる文字数が少ないため，頻繁に画面をスクロールさせる必要がある．これらの問題を解決するために，情報を携帯端末の 1 画面に収めるための文章要約や，文章の体言止め，言い換えなどの手法が研究され始めている[1)]．また，XML などの内部データをサーバ側に用意し，携帯端末の情報を得てそれぞれに適した HTML，CHTML などの表示データを動的に生成する方式も提案されている[2)]．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．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，作成者の意図に反した表示結果となることや，場合によっては表示不能になる問題が発生

† 東京電機大学工学部
School of Engineering, Tokyo Denki University
†† 東京大学情報基盤センター
Information Technology Center, The University of Tokyo
☆ 現在，株式会社豆蔵
Presently with Mamezou Co., Ltd.

(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
Fig. 1 An example of displaying a table on PC screen.

する[3]．

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，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．本アルゴリズムは，言語的性質を用いて表の各セル（表の 1 つのマス目）データに対するベクトルを用意し，セルの間の類似度をベクトル空間法によって算出する．算出したセルの類似度が低くなる部分には，内容的な切れ目があると認識する．この認識の結果を用いて表を項目名と項目データの組として携帯端末向けに表示変換し，被験者を用いて変換した結果を評価した．

以下，2 章では，携帯端末における表の表示の問題点をあげ，3 章では，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の類似度の定義と計算，およびアルゴリズムを提案し，実験的に評価する．4 章では，システムを用いて表を変換した結果と評価を示し，5 章でまとめを述べる．

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，具体的な問題点をあ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 Xiino[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．

図 2 AvantGo での表示例
Fig. 2 An example of displaying the same table on AvantGo.

図 3 Xiino での表示例
Fig. 3 An example of displaying the same table on Xiino.

図 2 の AvantGo 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 Xiino 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

図 4 スクロールしたときの Xiino での表示例
Fig. 4 An example of displaying the same table after turning the page.

図 5 rowspan オプションがあるページを表示した例
Fig. 5 An example of displaying a table with spans.

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．

2.2 Web ページ上の表の種類および型

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
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 つであり強調表現をしているものや，表として使用していないものなどがある．

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[7]．

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．

時間割型
行，列どちらにも項目名を持っている表である．

本研究では，これら 3 つの型に対応する構造認識および，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),10)] ので，次章において詳しく述べる．

3. 表の構造認識アルゴリズム

3.1 システムの概要

これまでに表を取り扱う研究として，プレインテキスト中の整形された表から構造を認識する研究[11)] があるが，正規化されている表を対象としており，複数の行または列に項目名を持つ表は対象としていない．表からの情報抽出[12)] でも同様に複数の行または列に項目名を持つ表は対象としていない．また，Web ページ中の表が本質的か，レイアウトとして使われているかを判別する研究[13)] がある．

本システムは，2.2 節で述べた本質的な表のみを対象とする．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．まず，3.2 節に示すように表の正規化を行い，3.3 節で述べるように表の各セルデータに対して言語的性質を適用したベクトルのマトリクスを作成する．次に行方向の切れ目を認識するために，各セルの列方向の類似度をベクトル空間法によって算出する．算出したセルの類似度に対して行ごとの平均値を求め，行の類似度が低くなる部

		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
Fig. 6 A table with rowspans and colspans.

		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
Fig. 7 Normalization of the table.

分には，行間に内容的な切れ目があると認識する．行と列を入れ換えて類似度を計算すれば，列間の切れ目を認識することができる．これについては，3.4 節で詳述する．最後に 3.5 節で述べるように，境界判定のための閾値を 10fold 交差検定で決定し，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** のように正規化する[14]．

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．

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

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 107 であり，内容は以下に示す．

3.3.1 ベクトルの各要素

- 連続する数値データ（1 次元）
  初期値から一定間隔で単純増加する連続性を持つ数字の行または列があれば，この要素を 1 とする．
- 句読点（6 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，句点，読点がないことをそれぞれ 1 つのベクトルの次元として定義する．ただし文字列中の出現位置は問わない．
  以下，文字については，1 バイトおよび 2 バイトの文字が存在する場合には各々を別の次元にする．
- 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（空白セル）”，“10 バイト以内”，“11 バイト以上”
  以上をそれぞれベクトルの次元とした．
- 接頭辞[15]（16 次元）
  “第”，“昭和”，“特”，“平成”，“要”，“(株)”，“貸付”，“☆”，“★”，“○”，“●”，“□”，“■”，“◇”，“◆”，1 文字として表現される “(株)”
  の 16 種の接頭辞に前方一致した場合に各々のベクトルの次元を割り当てる．
- 接尾辞[15]（45 次元）
  “位”，“日”，“点” ，“月”，“年”，“末”，“旬”，“発”，“(株)”，“課”，“学校”，“学科”，“局”，“系”，“空港”，“号”，“県”，“国”，“所”，“内”，“者”，“省”，“数”，“大学”，“中”，“貯金”，“調査”，“店”，“部”，“分野”，“曜日”，“鉄道”，“名”，“市”，“時”，“回”，“集”，“項”，“会”，“年度”，“分”，“券”，“科”，“温泉”，1 文字として表現される “(株)”
  の 45 種の接尾辞に後方一致した場合に各々にベクトルの次元を割り当てる．
- 単位（16 次元）
  “円”，“ドル”，“$”，“人”，“時”，“日”，“分”，“秒”，“個”，“票”，“ケース”，“kg”，“yard”，“feet”，“inch”
  の 16 種の単位が後方一致するか，アルファベットが部分一致した場合に各々にベクトルの次元を割り当てる．
- 特殊文字（13 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，
  “(”，“)”，“+”，“=”，“:”，“@”，2 バイトの “〜”
  の 13 種の各々にベクトルの次元を割り当てる．
- 文字種（5 次元）
  “ひらがな”，“カタカナ”，“漢字”，“数字”，“アルファベット” の 5 種の各々にベクトルを割り当てる．
- テーブルタグの属性（2 次元）
  colspan，rowspan のオプションがついているタグのセル，あるいは，colspan オプションがついているセルの次の行，rowspan オプションがついているセルの次の列は項目名となることが多い．よって，

“colspan オプションのセル内または colspan オプションがついているセルの次行”,
“rowspan オプションのセル内または rowspan オプションがついているセルの次列”
の 2 つを各ベクトルの次元に割り当てる．
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

3.4.1 ベクトルの計算

m 行 n 列の表の行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．

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

ここで，$\sum$ の範囲は，$k = 1, \cdots, m$，ただし，$k = i$ 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 を用いて類似度の計算方向を説明する．まず，行における項目名の境界を特定するために，各列のそれぞれのセルの間の類似度を求める．例として，1 行 1 列目，2 行 1 列目，m 行 1 列目の類似度の求め方を示す．図 8 (a) では，$Cell_{11}(1,1)$ を基準とし，$Cell_{11}(1,1)$ 以外で同列中のセルのコサイン値を計算し，求めたコサイン値の和の平均値を類似度 $Sim_{row}(1,1)$ とする．次に図 8 (b) では，$Cell_{12}(1,2)$ を基準とし，$Cell_{12}(1,2)$ 以外で同列中のセルとのコサイン値の和の平均を求める．そして，図 8 (c) に示すように $Cell_{m1}(m,1)$ まで同じ工程で類似度を計算する．このようにして順に最後の列まで計算を繰り返し，**図 9** のように類似度のマトリクスを構成する．次に，行の切れ目を特定するために，**図 10** のように，式 (3) を用いて行の類似度の平均を求める．

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

項目名を表す行と項目データを表す行とは類似度が低く，値も分散しているが，項目データを表す行どうしは類似度が高く，値がほぼ一定となる．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．実際に 2 行目と 3 行目の間が項目名と項目データの境界となる図 6 の表について Sim_{row} を計算し，その値の変化の様子を**図 11** に示した．

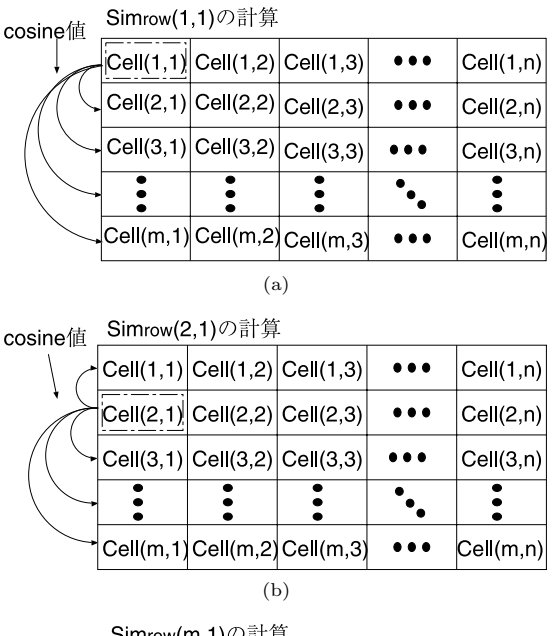

図 8 $Sim_{row}(m,1)$ の計算
Fig. 8 Calculation of $Sim_{row}(m,1)$.

Simrow(1,1)	Simrow(1,2)	Simrow(1,1)	● ● ●	Simrow(1,n)
Simrow(2,1)	Simrow(2,2)	Simrow(2,3)	● ● ●	Simrow(2,n)
Simrow(3,1)	Simrow(3,2)	Simrow(3,3)	● ● ●	Simrow(3,n)
⋮	⋮	⋮	⋱	⋮
Simrow(m,1)	Simrow(m,2)	Simrow(m,3)	● ● ●	Simrow(m,n)

図 9 類似度のマトリクス
Fig. 9 Matrix of similarities.

項目名の部分は類似度が低く，項目データの部分は値が一定となる．次に，類似度の比（$Sim_{row}(i)$ と $Sim_{row}(i+1), \cdots, Sim_{row}(m)$ の平均の比）$R(i)$ を式 (4) で定義する．$R(i)$ の比が小さければ i 行は項目名となり，大きくなれば項目データとなる．

$$R(i) = \frac{Sim_{row}(i)}{\frac{1}{m-i}\sum_{k=i+1}^{m} Sim_{row}(k)} \qquad (4)$$

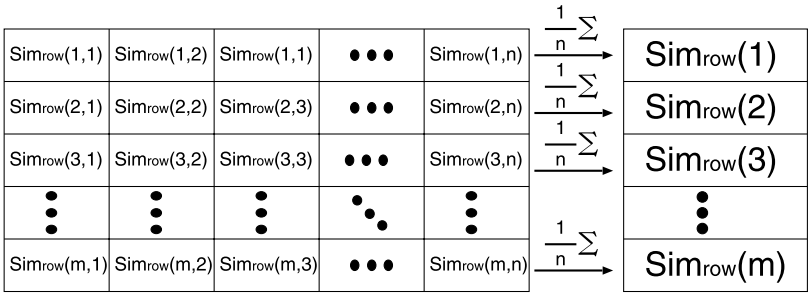

図 10 式 (3) の計算結果
Fig. 10 The result of calculation using Formula (3).

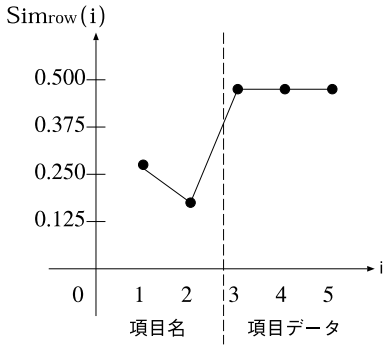

図 11 項目名と項目データの境界における Sim_{row} の変化の様子
Fig. 11 A characteristic of Sim_{row} following the number of rows.

式 (4) より求める値 $R(i)$ より，項目名と項目データの行の境界 T を次のアルゴリズムで求める．ただし，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 節の表の型にあてはめる．

縦一覧型

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．

横一覧型

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．

表 1 行方向の結果
Table 1 The result of lengthways.

データの種類	正解率
トレーニングデータ	83.23%
テストデータ	82.11%

表 2 列行方向の結果
Table 2 The result of sideways.

データの種類	正解率
トレーニングデータ	79.11%
テストデータ	78.11%

時間割型

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 節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．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 交差検定を用いる．Web 検索ロボットを用いて，<TABLE> タグを使用した本質的な表を含む Web ページを 2,193 収集し，その中からランダムに抽出した表を 300 用意した．本質的な表の条件としては，罫線を含み（BORDER 属性が 1 以上），2 行 2 列以上の大きさであることとした．最適な閾値 θ を求めるための教師データとして，この 300 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．

行方向の評価の結果を**表 1**，列方向の結果を**表 2** に示す．また，評価を行った表の大きさの平均は 9.2 行 6.3 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 の結果のうち，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 行目 1 列目が 22 個，2 行目 2 列目が 1 個である．評価に用いた表の行方向の $R(i)$ の平均値の分布を**図 12**

に，列方向の $R(i)$ の平均値の分布を**図 13** に示す．図 12，図 13 ともに，切れ目のない表の $R(i)$ は一定となり，1 行目（または列）で切れる表の $R(1)$ は低く，$R(2)$ 以降は一定となっている．2 行目（または列）で切れる表の $R(i)$ は傾向として右上がりであるが，差があまりみられない表が現れていることが分かる．3 行目で切れる表の $R(3)$ までの比は低く，$R(4)$ から値が高くなる．

表 1，表 2 の結果から，システムは行方向に 82%，列方向に 78%の正解率で表の項目名を認識することができる．認識を誤った表は，

(1) 実際には項目名の部分にもかかわらず，言語的類似度が高く切れ目を認識できない表（誤りの 40%）

(2) 項目データの部分にもかかわらず，言語的類似度が低いために切れ目を付けてしまった表（誤りの 60%）

の 2 つに大別できる．**図 14** は，行方向の切り分けは正しいが，言語的類似度が低いために列方向に誤って切れ目を付けてしまった表の例である．

次に，すべてのベクトルの要素が有効な場合の正解率を 100 として，ベクトルの要素が正解率に与える影響を相対的に評価した．実際にどのベクトル要素群がどの程度認識に影響を与えているのかを調べるために，言語的性質の各カテゴリごとにベクトルの要素群を無効にして正解率の変化を調査した．カテゴリは，1. 文字種，2. 接頭辞・接尾辞，3. 句読点・単位，4. 特殊文字，5. 文字長とした．その結果を**図 15** に示す．

この結果から，正解率に影響が大きいカテゴリは文字長であることが分かった．続いて文字種，接頭辞・接尾辞，句読点・単位，特殊文字の順となる．

表 3 交差検定によるテストデータとして評価をした 300 表の内訳
Table 3 The right boundaries of test data in 300 tables.

切れ目	行	列
0	70	183
1	202	115
2	25	2
3	3	0
合計	300	300

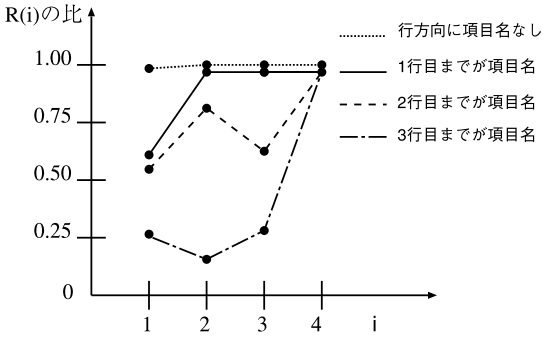

図 12 行方向での $R(i)$ の分布
Fig. 12 Distirbution of $R(i)$ of sideways.

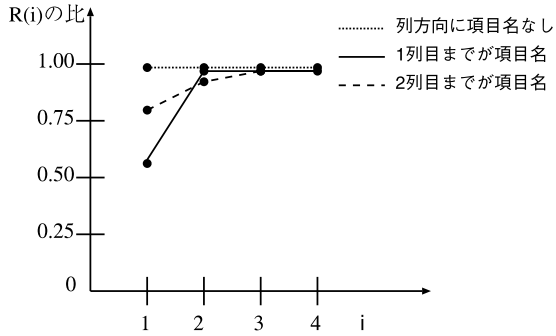

図 13 列方向での $R(i)$ の分布
Fig. 13 Distribution of $R(i)$ of lengthways.

年度	被保険者（加入者）数 (1)	受給者数				(2)／(1)
		老齢厚生年金		障害厚生年金	遺族厚生年金	
		老齢相当 (2)	通老相当			
平成（西暦）12 (2000)	百万人 34.3	百万人 8.7	百万人 5.4	百万人 0.3	百万人 3.5	% 25.2
13 (2001)	34.4	9.2	5.7	0.3	3.7	26.6
14 (2002)	35.0	9.7	6.0	0.3	3.8	27.5
15 (2003)	35.0	10.1	6.3	0.4	4.0	29.0
16 (2004)	34.9	10.6	6.6	0.4	4.2	30.3

(a) 人手で付けた正解位置

年度	被保険者（加入者）数 (1)	受給者数				(2)／(1)
		老齢厚生年金		障害厚生年金	遺族厚生年金	
		老齢相当 (2)	通老相当			
平成（西暦）12 (2000)	百万人 34.3	百万人 8.7	百万人 5.4	百万人 0.3	百万人 3.5	% 25.2
13 (2001)	34.4	9.2	5.7	0.3	3.7	26.6
14 (2002)	35.0	9.7	6.0	0.3	3.8	27.5
15 (2003)	35.0	10.1	6.3	0.4	4.0	29.0
16 (2004)	34.9	10.6	6.6	0.4	4.2	30.3

(b) システムが認識した結果

図 14 列方向の認識誤りとなった表の例
Fig. 14 An example of recognition failure of sideways.

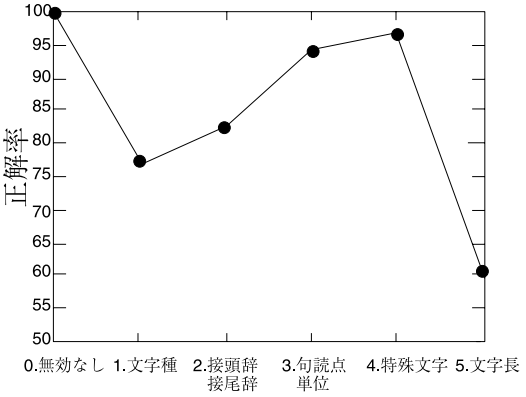

図 15 カテゴリの影響

Fig. 15 The effects to accuracy of each category in the vector.

図 16 図 1 の表をシステムで変換した例

Fig. 16 The result of transformation of the same table as Fig. 1.

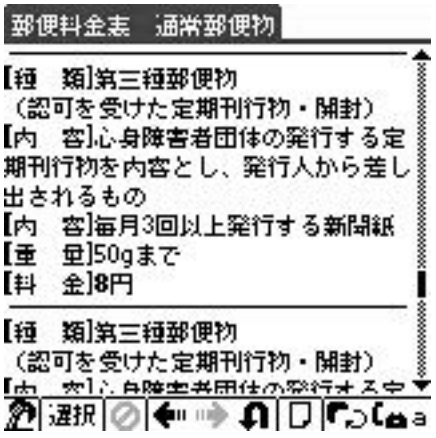

図 17 図 5 の表をシステムで変換した例

Fig. 17 The result of transformation of the same table as Fig. 5.

4. 携帯端末向け表示変換と検索時間による評価

4.1 携帯端末向け表示変換

2 章における考察の結果，携帯端末上に表を表示する際の問題は，項目名と項目データが乖離してしまうことが主な原因となっている．そこで，携帯端末上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，つねに項目名と項目データをペアで表示することにした．これには，3.5 節で認識した項目名と項目データを利用する．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．横方向の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，図 5 の表を変換した例を**図 16**，**図 17** に示す．図 16 では，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 17 では，スクロールによって，見えなくなってしまった項目名の “種類”，“内容”，“重量”，“料金” とそれらペアの値を表示することにより，表の最上部にページ戻すことなく値が何を示すのか理解できる．

4.2 検索時間による評価

携帯端末向け変換の評価のために，Palm OS Emulator[6] と Xiino を用いて表の中の特定の情報が得られるまでの時間を計測する実験を行った．被験者は 12 名で，用意した表に対して項目名を与えて，その項目データの情報を検索する課題を与え，情報を見つけるまでの時間を計測した．3.5 節に用いた表の中から 4 つの表を抽出し，それぞれ A，B，C，D とする．表の大きさは，A が 12 行 3 列，B が 30 行 5 列，C が 26 行 8 列，D が 34 行 5 列となっている．切れ目の位置は，A と C が 1 行 1 列目，B と D が 1 行目である．また，あらかじめシステムを使って変換した表を A'，B'，C'，D' とする．被験者には，A，B'，C，D' または A'，B，C'，D のいずれかの組（6 名ずつ）を与え，検索課題を提示した．検索開始から終了までを計測した平均時間を標準誤差とともに**図 18** に示す．

この結果から，小さい表を変換しても検索時間に大きな差はないが，表が大きくなると 20 秒以上の差がでていることが分かる．また，検索終了までの時間も変換していない場合は表の大きさに依存していることが分かる．したがって，表示領域が小さなデバイスにおけるデータの出力形式は 2 次元の表が必ずしも最適ではなく，項目名と項目データのペアの表示が有効であることが，本稿で述べた小規模な実験では実証さ

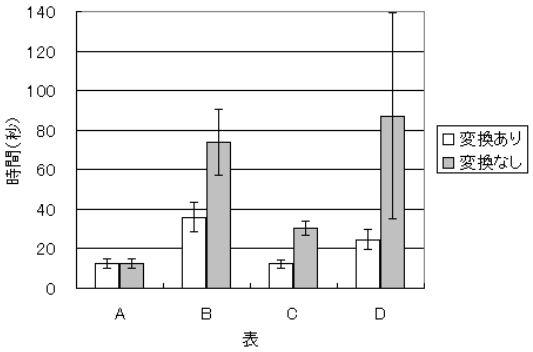

図 18 検索にかかった平均時間
Fig. 18 Average time of looking up the data in tables.

れた．

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行方向に 82%，列方向に 78%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，ベクトル要素の値の最適化を検討する．現段階では各々のベクトル要素の値は 1 か 0 としているため，類似度を計算する際に強く働いているものと強く働いていないものを同等に扱っている．ベクトルのカテゴリの影響の評価を基に，ベクトルの値を最適化し，項目名の認識率を向上させる．また，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したりするユーザインタフェースを実装する予定である．

謝辞 本研究は通信放送機構からの受託研究「モバイル環境における自然言語処理に関する研究」および，東京電機大学総合研究所研究費の支援による．

参 考 文 献

1) 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論文集，pp.33–41 (2002).
2) VertexLinkCorporation: C3GATEServer. http://www.vertexlink.co.jp/
3) 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，Vol.42, No.6, pp.576–581 (2001).
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：Xiino2.1/SJ. http://www.ilinx.co.jp/
7) Masuda, H., Yasutomi, D. and Nakagawa, H.: How to Transform Tables in HTML for Displaying on Mobile Terminals, *Proc. 6th NLPRS2001 Workshop of Automatic Paraphrasing: Theories and Applications*, pp.29–36 (2001).
8) Yoshida, M.: Extracting Attributes and Their Values from Web Pages, *Proc. 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 回年次大会講演論文集，pp.347–350 (2002).
10) 塚本修一，増田英孝，中川裕志：HTML の表形式データの変換と携帯端末表示への応用，情報処理学会研究報告 2002-NL-151，Vol.2002, No.87, pp.35–42 (2002).
11) Hurst, M. and Duglas, S.: Layout and Language: Preliminary Experiments in Assingning Logical Structure to Table Cells, *Proc. 5th Conference on Applied Natural Language Processing*, pp.217–220 (1997).
12) 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，人工知能学会誌，Vol.14, No.6, pp.69–77 (1999).
13) Wang, Y. and Hu, J.: A Machine Learning Based Approach for Table Detection on The Web, *Proc. 11th International World Wide Web Conference* (*WWW2002*), pp.242–250 (2002).
14) Chen, H.-H., Tsai, S.-C. and Tsai, J.-H.: MiningTables from Large Scale HTML Texts, *Proc. COLING2000*, pp.166–172 (2000).
15) 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講演論文集，pp.93–94 (2002).

(平成 14 年 12 月 20 日受付)
(平成 15 年 7 月 4 日採録)

（担当編集委員 大山 敬三）

増田　英孝（正会員）

1995 年東京電機大学大学院工学研究科電気工学専攻博士後期課程修了．博士（工学）．同年東京電機大学工学部電気工学科助手．2002 年より東京電機大学工学部情報メディア学科講師．ACM，言語処理学会各会員．

塚本　修一（学生会員）

2002 年東京電機大学工学部電気工学科卒業．現在，同大学大学院工学研究科情報通信工学専攻博士前期課程在学中．

安富　大輔

2000 年東京電機大学工学部電気工学科卒業．2002 年同大学大学院工学研究科電気工学専攻博士前期課程修了．現在，株式会社豆蔵勤務．

中川　裕志（正会員）

1975 年東京大学工学部卒業．1980 年同大学院博士課程修了．工学博士．1980 年より横浜国立大学工学部勤務．1999 年より東京大学情報基盤センター教授．自然言語処理の研究に従事．ACL Exective Commitee，言語処理学会副会長．