社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

WWW からの性能表抽出のためのキーワード獲得と重み付け

林　　晃司†　　嶋田　和孝†　　遠藤　　勉†

† 九州工業大学情報工学部知能情報工学科
〒 820-0052 福岡県飯塚市大字川津 680-4
E-mail: {k_haya,shimada,endo}@pluto.ai.kyutech.ac.jp

あらまし　我々は現在，Web 上の製品の性能などを記述した表を用いた製品選択支援システムの構築を行っている．Web 上の表は HTML の<TABLE> タグを用いて記述されるが，<TABLE> タグは表を記述する以外にも，レイアウトを整えたりする場合にも頻繁に用いられる．ある特定の領域においては，<TABLE> の 70% がレイアウト目的で使われているとの報告もある．そのため，HTML 文書中の<TABLE> タグが表なのか，それとも他の目的で使用されているのかを判別する必要がある．本論文では，Web からの性能表抽出のためのキーワード生成の手法について提案する．キーワードの重み付けには，エントロピーを用いた手法とベイズの定理を用いた手法の二つを検証した．実験により，それぞれの重み付けによる表抽出の精度について考察する．

キーワード　表抽出，キーワード獲得，重み付け

Keywords and Weighting for Product Specifications Extraction

Koji HAYASHI†, Kazutaka SHIMADA†, and Tsutomu ENDO†

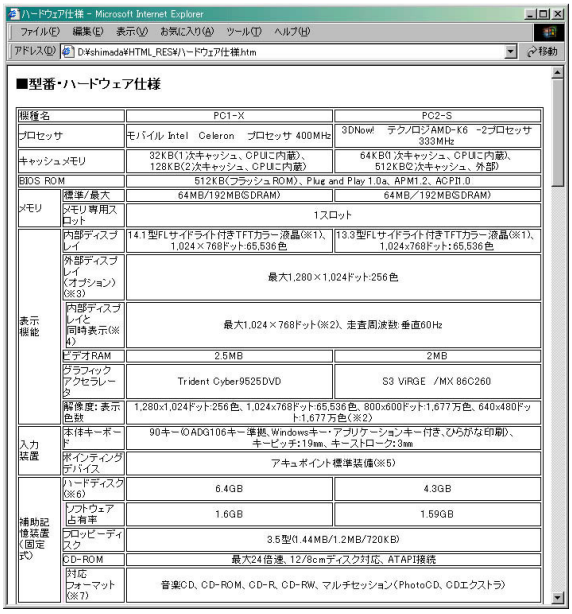
† Department of Artificial Intelligence Kyushu Institute of Technology,
680-4 Iizuka Fukuoka 820-0052 JAPAN
E-mail: {k_haya,shimada,endo}@pluto.ai.kyutech.ac.jp

Abstract　Product specifications contain many data. It is not, however, clear which is the characteristic data in them. We are developing a multi-specifications summarization system using extracted characteristic data from the product specifications. The specifications are written in a <TABLE> tag. The presence of the <TABLE> tag in an HTML document does not necessarily indicate the presence of specifications. Less than 30% of HTML <TABLE> tags are real tables in one particular domain. In this paper, we propose a method for keyword extraction for product specifications extraction. We evaluate the performance for two keyword sets, which are constructed by entropy and a Bayes theorem based method.

Key words　Table Extraction, Keyword extraction, Weighting

1. はじめに

近年のインターネットの急速な普及により，職場や家庭にいながら世界中から発信された情報にアクセスできる環境が整ってきた．これに伴い，紙面で伝えられていた情報の電子化が進んでいる．電子化されるようになった情報の 1 つとして，図 1に示すような製品の性能や機能などを記述した表（以下，製品性能表と呼ぶ．）が挙げられる．製品性能表には，その製品に関する様々な具体的データが載せられている．しかし，データが載せられているだけで，どの項目がその製品の特徴となるかは製品性能表を見ただけでは一般的には判断しにくい．あるユーザが PC を購入しようとしていると仮定すると，そのユーザは多くの PC メーカのサイトから製品性能表などを含んだ様々な情報を抽出し，比較する必要がある．しかしながら，多数の製品性能表から求めている製品を探し出すことは一般に労力がかかる．その理由としては，



■型番・ハードウェア仕様

機種名		PC1-X	PC2-S
プロセッサ		モバイル Intel Celeron プロセッサ 400MHz	3DNow! テクノロジAMD-K6 -2プロセッサ 333MHz
キャッシュメモリ		32KB(1次キャッシュ、CPUに内蔵)、128KB(2次キャッシュ、CPUに内蔵)	64KB(1次キャッシュ、CPUに内蔵)、512KB(2次キャッシュ、外部)
BIOS ROM		512KB(フラッシュROM)、Plug and Play 1.0a、APM1.2、ACPI1.0	
メモリ	標準/最大	64MB/192MB(SDRAM)	64MB/192MB(SDRAM)
	メモリ専用スロット	1スロット	
表示機能	内部ディスプレイ	14.1型FLサイドライト付きTFTカラー液晶(※1)、1,024×768ドット:65,536色	13.3型FLサイドライト付きTFTカラー液晶(※1)、1,024x768ドット:65,536色
	外部ディスプレイ(オプション)(※3)	最大1,280×1,024ドット:256色	
	内部ディスプレイと同時表示(※4)	最大1,024×768ドット(※2)、走査周波数:垂直60Hz	
	ビデオRAM	2.5MB	2MB
	グラフィックアクセラレータ	Trident Cyber9525DVD	S3 ViRGE /MX 86C260
	解像度:表示色数	1,280x1,024ドット:256色、1,024x768ドット:65,536色、800x600ドット:1,677万色、640x480ドット:1,677万色(※2)	
入力装置	本体キーボード	90キー(OADG106キー準拠、Windowsキー・アプリケーションキー付き、ひらがな印刷)、キーピッチ:19mm、キーストローク:3mm	
	ポインティングデバイス	アキュポイント標準装備(※5)	
補助記憶装置(固定式)	ハードディスク(※6)	6.4GB	4.3GB
	ソフトウェア占有率	1.6GB	1.59GB
	フロッピーディスク	3.5型(1.44MB/1.2MB/720KB)	
	CD-ROM	最大24倍速、12/8cmディスク対応、ATAPI接続	
	対応フォーマット(※7)	音楽CD、CD-ROM、CD-R、CD-RW、マルチセッション(PhotoCD、CDエクストラ)	

図 1　製品性能表の例

（1） 各サイトでは自社製品の特徴は述べられているが，他社製品などとの比較はあまりなされていない．

（2） 各サイトごとに様々な表現方法がある．

（3） 要求と製品の特徴を関連づけるには，その製品に対してある程度知識が必要である．

などが挙げられる．ユーザの要求を満足させるには，複数のサイトから情報を抽出し，統合する必要がある．また，各々の製品の相対的な特徴正しく抽出できたとしても，表を提示するだけでは，ユーザにとって読みやすい情報であるとは限らない．

我々は，複数の製品性能表を解析し，各々の製品データを比較することで特徴データを抽出し，その特徴データを用いたランキング，文章や表，グラフといった複数の形式で出力する製品選択支援システムの開発を進めている [2] [8]．図 2に本システムの概要を示す．システムは HTML で記述された製品性能表を各メーカのサイトから抽出し，それらを解析することで表を表構造へ変換する．得られた表構造中の数値データや文字データを比較し，それぞれの製品の相対的な特徴データを抽出し，ユーザの要求に応じてスコア付けを行う．その特徴データを基に文章生成と表の再構成，グラフ生成を行ない，複数の形式を統合した要約を出力する．図 3は我々が開発した製品選択支援システムである．

本稿では，この製品選択支援システムの性能表抽出処理のためのキーワード獲得とその重み付け処理について提案する．重み付けは，エントロピーによる手法とベイズの定理を用いた手法の二種類について行い，その精度を比較・考察する．図 4は本稿で述べる性能表抽出処理の流れである．

2. 関 連 研 究

従来の表解析に関する研究は，文書イメージ [4] やプレインテキスト [6] を扱ったものが多い．ネットワークの普及により， Web 上の表を扱っている研究も多く存在する [1] [10] [11]． HTML 文書では，表は<TABLE> タグを用いて記述される．しかし，この<TABLE> タグは文書のレイアウトを整えるためにも頻繁に使われ，ある特定の領域では，<TABLE> タグが表として用いられているのは全体の 30% 程度であるとの報告もある [1]． Chen ら [1] の研究では，航空会社の表を対象とした Web 上からの表抽出・解析手法が述べられている．表抽出は，人手で作成されたルールやヒューリスティックなどによって行われている．これらのルールを対象となるドメイン毎に作り替えることはコストがかかり，また，作成者にある程度の専門的知識も必要となるなどの問題がある．機械学習などを用いた表抽出の研究としては， Wang ら [10] や Yoshida ら [11] による研究がある．Wang ら [10] の研究では，表のレイアウト的な情報や表中に存在するコンテンツ（画像があるか文字があるかなど）の情報を素性とし，決定木とサポートベクターマシンの二種類の機械学習の手法について比較・考察している． Yoshida ら [11] は， EM アルゴリズムを用い，表の認識やクラスタリングを行う手法を提案している．これらの研究の目的は，<TABLE> タグで囲まれた領域が，本当に表であるかどうかを判定することである．我々の目的は，現在構築中の製品選択支援システムの入力データの自動獲得であり，<TABLE> タグで囲まれた場所が表かそうでないかの判定ではなく，その領域が性能表であるかどうかを判定することである．

3. キーワード抽出と重み付け

まず，企業のサイトから性能表を抽出する必要がある．これらのサイトからの Web ページ抽出は既存のダウンローディングソフトを用いている．本節では，まずキーワード候補の抽出処理について述べる．続いて，得られたキーワード候補に対する重み付け処理について述べる．重み付け手法としては，エントロピーを用いる手法とベイズの定理を用いる手法の二種類がある．

3.1 キーワード抽出

<TABLE> で記述された領域が表であるかどうかを判別するために，我々は表判別のためのキーワードを作成した．キーワードの定義は

（1） 性能表の項目欄中に出現する単語（<TD>～</TD>）

（2） 一定長以内の文章中に出現する単語

（3） 性能表が存在するページ内で顕著または限定的

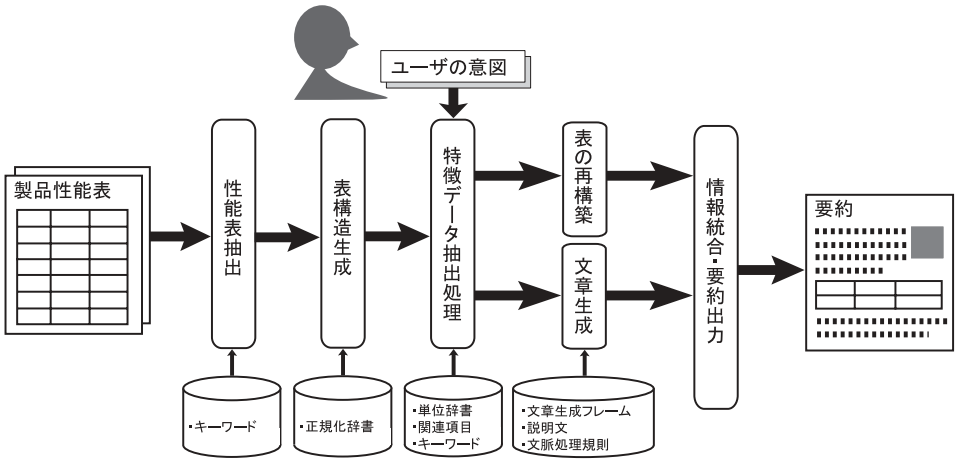


図 2　システムの概要

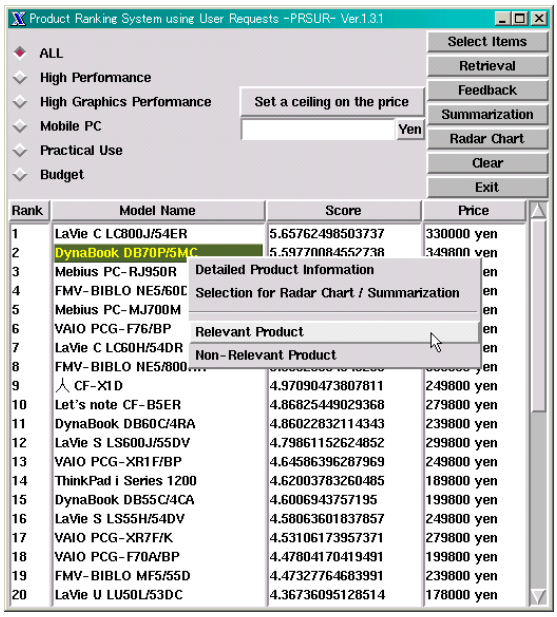


図 3　製品選択支援システム

に出現する単語

とする．この定義に基づき，HTML 文書から性能表抽出のためのキーワード候補を抽出する．実際の抽出処理は以下の手順で行われる．

（1） HTML 文書から<TABLE> タグで記述された領域を抽出する．

（2） <TABLE> タグ中の各<TR> タグ中の初めの<TD> タグの内容を抽出する．

（3） 得られた文字列が 25 文字以内であれば，形態素解析を行い，キーワード候補を抽出する．

形態素解析には奈良先端科学技術大学で開発された「茶筌」[5] を用いた．

3.2　エントロピーによる重み付け

得られたキーワード候補に対して重みを計算し，その重みを基にキーワードを抽出する．情報理論的な観点から語の特定性を考えれば，ある単語が文書集合中の各

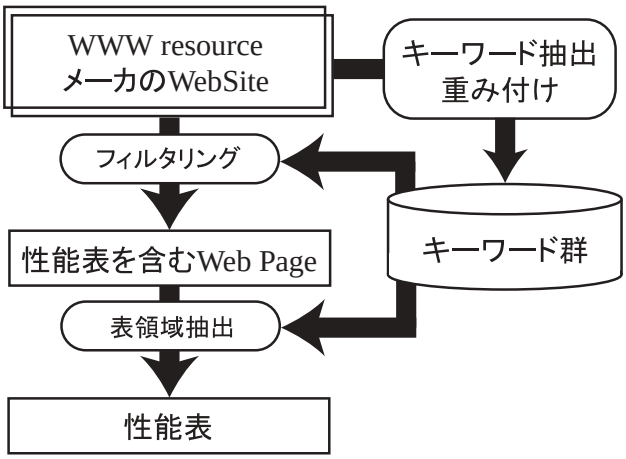


図 4　性能表抽出の流れ

文書にどれだけ偏って出現するかを情報理論のエントロピーの考え方を用いて数値化することができる．その数値を基にキーワード候補からキーワードを抽出する．キーワード抽出の手順を以下に示す．

（1） HTML 文書群 $D = \{d_1, ..., d_N\}$ からキーワード候補を抽出する．

（2） キーワード候補 t の各文書 d における頻度 $tf(t,d)$ を計算する．

（3） 文書群 D を性能表が存在する文書群 D_{real} とそうでない文書群 D_{no} に分割する．

（4） 文書群 D_{real} におけるキーワード候補 t の重み wr_t^{real} と文書群 D_{no} におけるキーワード候補 t の重み wr_t^{no} を以下の式で計算する．

$$wr_t^{real} = \frac{w_t^{D_{real}}}{w_t^{D_{no}}}, \quad wr_t^{no} = \frac{w_t^{D_{no}}}{w_t^{D_{real}}}.$$

ここで，

$$w_t^{D_n} = \log \sum_{k=1}^{M} tf(t,k) + \sum_{i=1}^{M} \frac{tf(t,i)}{\sum_{j=1}^{M} tf(t,j)} \log \frac{tf(t,i)}{\sum_{j=1}^{M} tf(t,j)}$$

であり，M は文書群 D_{real} もしくは D_{no} における文

書の総数である．

ここで，性能表を含む文書中の<TABLE>タグ内に顕著に存在する語をキーワードと呼ぶ．キーワードの重みは，$ws_t^{real} = df(t, D_{real}) \times wr_t^{real}$ で求める．一方，性能表を含まない文書中の<TABLE>タグ内に顕著に存在する語をノイズワードと呼ぶ．ノイズワードの重みは，$ws_t^{no} = df(t, D_{no}) \times wr_t^{no}$ で求める．ここで，$df(t, D_{real})$ および $df(t, D_{no})$ は，文書群 D_{real} もしくは D_{no} 中の単語 t を含む文書の数である．

3.3 ベイズの定理による重み付け

ベイズの定理はパターン認識・分類の世界でよく知られた確率ベースの手法である [7]．$C = [C_j]_{j=1}^{M}$ において，$P(C_j)$ （ $\sum_{j=1}^{M} P(C_j) = 1$ ）は事前確率と呼ばれる．事前確率と条件付き確率密度分布 $p(x|C_j)$ （ $\int p(x|C_j)dx = 1$ ）が事前に得られる場合，単語 x がクラス C_j に属する事後確率 $P(C_j|x)$ は次の式で求められる．

$$P(C_j|x) = \frac{P(C_j)p(x|C_j)}{p(x)}$$

ここで，

$$p(x) = \sum_{j=1}^{M} P(C_j)p(x|C_j), \qquad \int p(x)dx = 1$$

であり，

$$\sum_{j=1}^{M} P(C_j|x) = 1$$

となる．ここで，$C = \{D_{real}, D_{no}\}$ である．すべての単語に対して，各クラスでの事後確率を求める．我々は上記の式で，ある単語 t を考えたとき，得られる事後確率 $P(C_j|t)$ をその単語の重みとする．すなわち，ある単語 t のキーワードとしての重みは，$ws_t^{real} = P(D_{real}|t)$ であり，ノイズワードとしての重みは，$ws_t^{no} = P(D_{no}|t)$ となる．

4. 性能表抽出処理

図 4で示したように，抽出処理は，フィルタリングと表領域抽出の二つのプロセスからなる．

4.1 フィルタリング

まず，初めに HTML 文書のフィルタリングを行う．ここで，フィルタリングとは，全 HTML 文書から性能表を含む HTML 文書のみを抽出することを指す．フィルタリングでは，前節で獲得されたキーワードとノイズワードを用いて，性能表を含む HTML 文書を抽出する．フィルタリングの流れを以下に示す．

（1） HTML 文書 d_i から<TABLE>タグで記述された領域を抽出する．

（2） <TABLE>タグ中の<TD>タグの内容を抽出する．

（3） 次の式を計算する．

$$Ratio_{real} = \frac{\text{ヒットしたキーワードの数}}{\text{キーワードの総数}}$$

$$Ratio_{no} = \frac{\text{ヒットしたノイズワードの数}}{\text{ノイズワードの総数}}$$

（4） 得られた値に対して次の式を計算する．

$$Score_i = Ratio_{real} \times \frac{Ratio_{real}}{Ratio_{no}}$$

（5） $Score_i$ が閾値 $th1$ 以上であれば，その HTML 文書 d_i に性能表が含まれているとして抽出する．

4.2 表領域抽出

続いて，表領域の抽出処理について述べる．表領域抽出とは，フィルタリングによって抽出された HTML 文書の中から性能表である<TABLE>タグ領域を抽出することを指す．表領域抽出は，キーワードのみで処理される．表領域抽出処理の手順を以下に示す．

（1） HTML 文書 d_i から<TABLE>タグで記述された領域 $table_j$ を抽出する．

（2） <TABLE>タグ中の<TD>タグの内容を抽出し，キーワードが一つも含まれていない場合はその領域を破棄する．

（3） $table_j$ に対して，その領域中に存在するキーワードの重みの総和 $Sum_j = \sum_{t \in K} ws_t^{real}$ を計算する．ここで，K はキーワード群である．

（4） Sum_j が最大になる領域 $table_j$ について，Sum_j の値が閾値 $th2$ 以上であれば，その領域 $table_j$ を d_i に存在する性能表として抽出する．ここで，

$$th2 = \frac{\text{キーワードの重みの総和}}{2}$$

である．

5. 実験・考察

製品性能表の抽出処理を評価するために 5 つのサイトから 200 文書を抽出した．このうち，性能表を含む文書の数は 100，含まない文書の数は 100 である．また，性能表を含む 100 文書は性能表だけで構成されているわけではなく，文書中に文字や画像，性能表ではない表も含んでいる．これらの 200 文書のうち，100 文書をキーワード抽出と重み付けアルゴリズムのための訓練データとして用いた．訓練データは性能表を含む 50 文書と含まない 50 文書で構成されている．実験はフィルタリングと表領域抽出処理について行った．

まず，HTML 文書のフィルタリングについて述べる．評価尺度としては，情報検索などの分野でよく用いられる再現率，適合率および F値 を用いた．

$$\text{再現率}(R) = \frac{\text{正しく抽出された}HTML\text{文書の数}}{\text{性能表を含む}HTML\text{文書の数}}$$

$$\text{適合率}(P) = \frac{\text{正しく抽出された}HTML\text{文書の数}}{\text{抽出された全ての}HTML\text{文書の数}}$$

$$F\text{値} = \frac{1}{\alpha\frac{1}{P} + (1-\alpha)\frac{1}{R}}$$

α は適合率と再現率の相対的な重みを表す．一般的にはこの α を 0.5 として扱う．これは再現率と適合率を同じに扱うことを意味する．しかし，フィルタリングは，HTML 文書から表領域を特定し，性能表を抽出する際の前処理である．つまり，適合率よりも再現率が重視される．そこで，ここでは $\alpha = 0.4$ として，F値 を計算する．これにより，F値 は再現率重視の評価値となる．

エントロピーを用いたキーワードとノイズワードおよび重みによる実験結果を表1および表2に示す．続いて，表3および表 4は，ベイズの定理を用いて作成されたキーワードおよびノイズワードの重みの場合である．表中の M_{real} はキーワードの数，M_{no} はノイズワードの数を表す．$th1$ はフィルタリング処理で用いる閾値である．表 2のキーワード数およびノイズワード数は，エントロピーを用いた重み付けによって得られた全キーワードおよびノイズワードの数である．表 3のキーワードは，$P(D_{real}|t) > 0.75$ かつ性能表を含む文書群 D_{real} の半分以上の文書に出現した語，ノイズワードは，$P(D_{no}|t) > 0.75$ かつ性能表を含まない文書群 D_{no} の $\frac{1}{10}$ 以上に出現した語である．表 4のキーワードは，$P(D_{real}|t) > 0.75$ かつ性能表を含む文書群 D_{real} の $\frac{1}{10}$ 以上に出現した語，ノイズワードは，$P(D_{no}|t) > 0.25$ かつ性能表を含まない文書群 D_{no} の $\frac{1}{10}$ 以上に出現した語である．ベイズの定理を用いた場合の W_1 および W_2 は重みの種類を示しており，W_1 は， 3.3 節で示したベイズの定理で得られた確率そのものを重みとして用いたものである．W_2 はある単語 t のキーワードとしての重みをノイズワードの重みで割ったもの，およびその逆を計算した場合に得られる値を重みとしたものである．すなわち，W_1 においては，

$$ws_t^{real} = P(D_{real}|t) \qquad ws_t^{no} = P(D_{no}|t)$$

であり，W_2 においては

$$ws_t^{real} = \frac{P(D_{real}|t)}{P(D_{no}|t)} \qquad ws_t^{no} = \frac{P(D_{no}|t)}{P(D_{real}|t)}$$

である．

実験結果により，エントロピーを用いた手法よりもベイズの定理を用いた手法の方が高い再現率および適合率を得た．ベイズの定理を利用した場合，W_2 のように重みの範囲を 0 〜∞に拡張するよりも 0 〜 1 とした W_1 の場合の方が良い結果を得られた．エントロピーによる手法も理論的には重みの範囲は 0 〜∞であり，キーワードおよびノイズワードの重みはある程度の範囲に正規化された値の方が良い結果を得られることが確認された．エントロピーによる手法では，キーワードおよびノイズワードを全て使った場合の方が若干精度がよいが，有意な差であるとはいえず，キーワードおよびノイズワードを追加することの有意性は低い．ベイズの定理を利用した手法では，むしろキーワードおよびノイズワード数を増加させた方が全体の精度が落ちる傾向がみられた．これにより，提案した手法，特にベイズの定理を用いた手法は少ないキーワードおよびノイズワードで，性能表を含んだ HTML 文書のフィルタリングが行えることが確認された．

表1　エントロピー $M_{real} = 30, M_{no} = 15$

閾値 $th1$	再現率	適合率	F値
0.20	100.0%	80.7%	91.3%
0.25	100.0%	81.3%	91.6%
0.30	95.0%	82.6%	89.6%
0.35	92.0%	82.1%	87.8%

表2　エントロピー $M_{real} = 56, M_{no} = 37$

閾値 $th1$	再現率	適合率	F値
0.20	100.0%	80.0%	90.9%
0.25	100.0%	82.0%	91.9%
0.30	100.0%	85.5%	93.6%
0.35	96.0%	88.1%	92.7%

表3　ベイズの定理 $M_{real} = 19, M_{no} = 15$

閾値 $th1$	重み	再現率	適合率	F値
0.20	W_1	100.0%	92.6%	96.9%
	W_2	100.0%	90.9%	96.2%
0.30	W_1	100.0%	94.3%	97.7%
	W_2	100.0%	95.2%	98.0%
0.40	W_1	100.0%	98.0%	99.2%
	W_2	99.0%	97.1%	98.2%
0.50	W_1	100.0%	99.0%	99.6%
	W_2	99.0%	98.0%	98.6%
0.60	W_1	98.0%	98.99%	98.4%
	W_2	99.0%	98.0%	98.6%

続いて，表領域抽出処理について述べる．フィルタリングによって得られた結果を用いて表領域を抽出した．エントロピーを用いた手法では，$M_{real} = 30$ および $M_{no} = 15$ でもっともフィルタリングの精度が良かった閾値 $th1 = 0.25$ の実験結果を用いた．同様にベイズの

表 4　ベイズの定理 $M_{real} = 66, M_{no} = 41$

閾値 $th1$	重み	再現率	適合率	F値
0.20	W_1	100.0%	97.1%	98.8%
	W_2	100.0%	95.2%	98.0%
0.30	W_1	92.0%	97.9%	94.3%
	W_2	92.0%	96.8%	93.9%
0.40	W_1	83.0%	97.65%	88.3%
	W_2	86.0%	98.9%	90.7%
0.50	W_1	76.0%	100.0%	84.1%
	W_2	78.0%	98.7%	85.2%
0.60	W_1	57.0%	100.0%	68.8%
	W_2	63.0%	100.0%	73.9%

表 5　表領域抽出処理　- エントロピー -

閾値 $th1$	再現率	適合率
0.25	93.0%	96.9%

表 6　表領域抽出処理　- ベイズの定理 -

閾値 $th1$	再現率	適合率
0.50	95.0%	100.0%

定理を用いた手法では，$M_{real} = 19$ および $M_{no} = 15$ で閾値 $th1 = 0.50$ の実験結果を用いた．表領域の特定処理においても，ベイズの定理を用いた手法の方が，高い再現率および適合率を得た．エントロピーを用いた手法およびベイズの定理を用いた手法の双方とも適合率に比べ，再現率が若干低い．これは表領域抽出処理で用いた閾値 $th2$ の決め方に問題があるためだと考えられる．この閾値 $th2$ の決定法は今後の課題の一つである．

総じて，フィルタリングにおいても，表領域抽出処理においても，高い再現率および適合率を得ることができた．性能表抽出処理で用いるキーワードおよびノイズワードの追加は精度向上には有効ではなく，少ないキーワードおよびノイズワードで実現できることが確認された．ベイズの定理においては，フィルタリングでほぼ 100% の F値 を得た．実験により，提案した手法の有効性が確認された．

6. おわりに

本稿では，現在構築中の製品選択支援システムの入力部にあたる，Web 上からの製品性能表抽出処理のためのキーワード獲得手法とその重み付け処理について述べた．キーワードの重み付けは，エントロピーを用いた手法とベイズの定理を用いた手法で行い，その精度を比較した．フィルタリングでは，ベイズの定理を用いた手法により，ほぼ 100% の F値 を得た．表領域抽出処理においても，エントロピーを用いた手法，ベイズの定理を用いた手法の双方とも高い再現率と適合率が得られ，提案手法の有効性が確認された．また，抽出処理全体が，比較的少ないキーワードおよびノイズワードで実現できることも実験により確かめられた．

現在，エントロピーによる手法を他種類の製品の表に移行した際の精度について検証している [3]．同様にベイズの定理を用いた手法が他種類の製品の表にどれだけ有効であるかの評価を行う必要があるだろう．また，提案した二手法以外の機械学習アルゴリズムを用いたキーワード獲得および重み付けなども今後の課題の一つである．表領域抽出処理では，若干再現率が適合率を下回る傾向が見られた．これは，表領域抽出処理で用いる閾値の決め方に問題があるためだと考えられる．より高い再現率および適合率を性能表抽出処理で得るためには，閾値の決め方も含めた表抽出処理の改良が必要であり，これも今後の課題に挙げられる．

文　献

[1] H. H. Chen, S. C. Tsai and J. H. Tsai: Mining tables from large scale HTML texts, Proc. of COLING2000, pp.166-172, 2000.

[2] A. Fukumoto, T. Endo and K. Shimada: Information extraction from specifications on the world wide web, Proc. of PACLING 2001, pp.109-116, 2001.

[3] 林晃司，嶋田和孝，遠藤勉: WWW からの製品性能表抽出，第 9 回言語処理学会年次大会， NLP9， 2003.

[4] J. Hu, R. Kashi, D. Lopresti, and G. Wilfong: Medium-independent table detection, Proceedings of Document Recognition and Retrieval VII, pp.23–28, 2000.

[5] 松本裕治，北内啓，山下達雄，平野喜隆，松田寛，高岡一馬，浅原正幸: 日本語形態素解析システム「茶筌」 http://chasen.aist-nara.ac.jp/index.html

[6] H.T. Ng, C.Y. Lim, and J.L.T. Koo: Learning to recognize tables in free text, Proceedings of the 37th Annual Meeting of ACL, pp.443–450, 1999.

[7] M. Pazzani and D. Billsus. Learning and revising user profiles: The identification of interesting web sites, Machine Learning 27, pp.313-331, 1997.

[8] 嶋田和孝, 遠藤勉: 特徴化された表データからの要約文生成処理, 信学技報, TL99-29, pp.25-31, 1999.

[9] 徳永建伸: 情報検索と言語処理, 言語と計算 5, 東京大学出版, 1999 .

[10] Y. Wang and J. Hu: A machine learning based approach for table detection on the Web, Proc. of The Eleventh International World Web Conference, 2002.

[11] M. Yoshida, K. Torisawa and J. Tsujii: Extracting ontologies from World Wide Web via HTML tables, Proc. of PACLING 2001, pp.332-341, 2001.