ユーザプロファイルに基づくビューページの動的生成によるWWW閲覧支援

品川徳秀[†] 北川博之[††] 川田純[†††]

近年，WWWの急速な普及，発展にともない，大量の情報がアクセス可能となったが，その反面，ユーザが必要とする情報の発見，利用が難しくなっている．一般に，ユーザがWWW上で必要な情報を探索するうえでは検索と閲覧の両者が必要である．本稿では，ユーザが必要とするWWWページ中の記述の特定を効果的に支援するためのユーザ視点に基づいた閲覧支援手法を提案する．本手法では，閲覧時にWWWページを解析し，ユーザの興味を反映した仮想的なWWWページ（ビューページ）を動的に生成・提供する．各ビューページは，元のページの論理構造を抽出し，それに基づきユーザの興味に適合すると思われる部分のみを抜き出すことで与えられる．これにより，ユーザの視点からWWWを閲覧することが可能になり，興味に合致する記述の発見を容易にする．本稿では，ビューページの生成手法と本手法を実装したプロトタイプシステムについて述べる．また，本手法の有効性に関する評価実験の結果もあわせて示す．

Dynamic Generation and Browsing of WWW View-pages Based on User Profiles

Norihide Shinagawa,[†] Hiroyuki Kitagawa[††] and Jun Kawada[†††]

Due to the recent and rapid advance of the WWW technology, a huge amount of information is available via the Internet. However, it is difficult for users to find and utilize relevant information. Generally, querying and browsing are both indispensable to access to the required information. In this paper, we propose a scheme to support the browsing phase of the information search based on specification of the user's interest. In our scheme, each user is given virtual WWW pages named view-pages reflecting his/her interest. Each view-page is dynamically generated based on the logical document structure extracted from the original page. Our approach makes it possible to browse the WWW pages from his/her viewpoint and to identify required information more easily. We also show how such a scheme can be implemented in a non-intrusive way in the current WWW browsing environment. Moreover, we show the result of an experimantal evaluation of our scheme.

1. はじめに

近年，WWWの急速な普及，発展にともない，大量の情報が利用可能となったが，その反面，ユーザが必要とする情報の発見，利用が難しくなっている．このような問題に対し，WWWにおける探索やその閲覧の支援を行うための様々な研究がなされている[1)~4)]

WWW上で必要とする情報を記述したページを探すためには，検索と閲覧の両者が必要である．通常，ユーザはまず検索エンジン等に適当な問合せを行い，閲覧対象となるページの候補を絞り込む．続いて，候補ページを選択・閲覧し，必要ならリンクをたどったり，問合せ結果の候補ページ群から別の候補ページを選択したりする．また，場合によっては問合せ条件を変更し，再度問合せからやり直すこともある．このように，ユーザの欲する情報が得られるまで検索と閲覧が繰り返される．

ここで，ユーザの興味は検索時には問合せ条件として明示的に表現されるのに対し，閲覧時には利用されていないことに注意する必要がある．このため，ユーザがいかなる興味を持っていようとも，ユーザに提供される情報にはその視点は反映されず，各ページはつ

† 筑波大学工学研究科
Doctoral Program in Engineering, University of Tsukuba
†† 筑波大学電子・情報工学系
Institute of Information Sciences and Electronics, University of Tsukuba
††† 筑波大学システム情報工学研究科
Graduate School of Systems and Information Engineering, University of Tsukuba

ねに同じ様式で表示が行われる．すなわち，そのページのどの部分に有用な記述がなされているかを判断するための手がかりのないまま，ユーザがそれを特定する作業を行う必要がある．しかし，WWW の閲覧では多くの未知のページを扱わなければならず，なかには記述量の多いページも含まれる．このような状況において，それらの構造や内容を理解し，有用な記述部分を特定することは決して容易なことではない．また，既存の WWW ブラウザでは文字列検索機能を提供しているが，そのような単純な文字列検索では，ユーザを十分に支援することは困難であり，つねに同じ表示しか得られないという状況は改善されない．

我々は，このような背景の下に，ユーザの視点に基づいた WWW 閲覧支援手法の研究を進めてきた[5),6)]．本手法は，ユーザの視点を記述したユーザプロファイルを利用し，WWW ページの閲覧時に**ビューページ**を動的に生成・表示を行う．ビューページは，本質的にはユーザプロファイルとして与えられたユーザの興味と提示内容の詳細度に基づいて生成された要約である．その生成の際，HTML 文書中の論理的な構造を**論理木**としてモデル化して利用する．各ページに対して自動的にビューページが生成・提供されることによって，ユーザは視点によって異なったビューページを与えられることになる．また，通常の WWW 環境において本手法を実装したプロトタイプシステムを開発した．本プロトタイプシステムにおいては，ユーザは特別なブラウザを使うことなく，現在利用している WWW ブラウザで本手法の提供する環境を利用可能である．さらに，本稿では，本手法の有効性についての実験評価をあわせて示す．

以下が本稿の構成である．まず，2 章で本手法の概要を示す．次に，3 章で論理木を導入し，その抽出手法を述べる．4 章ではユーザプロファイルに基づいたビューページの生成方法について説明する．5 章ではプロトタイプシステムについて述べる．6 章では本手法の有効性に関する実験評価を示す．7 章では関連研究について言及する．最後に，8 章で本稿のまとめを行う．

2. 概　　要

本章では，本手法の概要を示す．**図 1** に本アプローチを図示する．

本環境では，ユーザの視点を記述したユーザプロファイルを利用する．ユーザプロファイルは，ユーザの興味を記述したキーワード群と表示の詳細度を制御する閾値から構成される．ユーザが WWW ページへアクセスすると，ユーザプロファイルに基づいてビューページが動的に生成され，ブラウザに表示される．このビューページにハイパーリンクが含まれているとき，ユーザは通常のページと同様にそれをたどることができ，リンク先のページについてもビューページが生成される．このようにビューページの提供が暗黙的に行われることで，ユーザ独自の視点から WWW 空間を探索可能となる．各ビューページは本質的にはユーザの視点に基づいた要約であり，その生成は次のように行われる．対象とする WWW ページは，HTML で記述されているものとする．

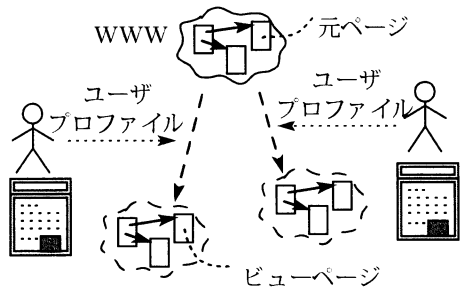

図 1 本手法のアプローチ
Fig. 1 Overview of the proposed scheme.

(1) 初めに，WWW ページ中から文書の論理構造を抽出し，論理木としてモデル化する．論理木は HTML の一部のタグに着目して導出される．

(2) 続いて，論理木の各ノードとユーザプロファイル中で与えられるキーワード群との類似度を示すスコアを計算する．そして，スコアがユーザプロファイル中で与えられる閾値よりも低いノードをマークする．最後に，すべての上位ノードと内部ノードがマークされた部分木を取り除いた木を生成し，それから HTML 文書を生成することでビューページを生成する．

(1) の詳細は 3 章で，(2) の詳細は 4 章で述べる．

3. 論　理　木

本章では，HTML 文書からの論理木の導出方法について説明する．まず，その際に利用する HTML タグの分類を 3.1 節で与え，3.2 節で具体的な論理木の導出方法を説明する．

3.1 HTML タグの分類

まず，HTML タグを以下の 4 グループに分類する．これは，3.2 節で示す論理木の導出と，4 章で示すノードのスコアの計算に用いられる．

(a) 構造主体のタグ　このグループには，主に見出し，段落，引用，リスト等の文書構造を示すために用いられる次のタグが含まれる：`H1`，`H2`，`H3`，`H4`，`H5`，`H6`，

P, DIV, BLOCKQUOTE, UL, OL, DL, TABLE. これらは，HTML 文書から論理構造を抽出する際の手がかりを与える．このことから，これらを 3.2 節で説明する論理木の導出に利用する．

(b) 表示主体のタグ このグループには，主に特殊な表示効果を得るために用いられる次のタグが含まれる：STRONG, EM, TT, I, U, B, BIG, SMALL, STRIKE, S, FONT, DL. これらは文書の論理構造を示すものではなく，主に重要部分を強調するために用いられる．このことは，4.1 節で説明するように，論理木のノードの特徴ベクトルを導出する際に利用される．

(c) メディア主体のタグ このグループには，画像やアプレットといったメディアオブジェクトを埋め込むために用いられる次のタグが含まれる：IMAGE, FORM, SCRIPT, APPLET, OBJECT, EMBED, MAP. これらのタグは論理木の導出には影響を与えず，また，その内容は特徴ベクトルを求める際にも無視される．

(d) その他のタグ 上記以外のタグは特別な効果を持たず，これらで囲まれた文書要素は通常の文字列として扱われる．すなわち，論理木の導出に影響を与えず，特徴ベクトルを求める際にも特別な重みを持たない．

3.2 論理木の導出

本節では HTML 文書からの論理木の導出法について述べる．一般に，HTML のタグで与えられる階層構造は非常に平坦であり，文書中の記述の意味的なまとまりとして用いられる章や節といった論理構造を直接的には反映しない．このため，適切なビューページの生成を行うためには論理構造の抽出が必要である．構造主体のタグは，この導出において重要な手がかりを与える．たとえば，H1, ..., H6 は各レベルの節の見出しを与える[2]．

論理木は次の 9 種のノードからなる：$\ll doc \gg$, $\ll desc(L) \gg$, $\ll leading(L) \gg$, $\ll trailing(L) \gg$, $\ll packed(L) \gg$, $\ll block(L) \gg$, $\ll heading(L) \gg$, $\ll paradiv \gg$, $\ll paragraph \gg$. ここで，L は構造の階層の深さのレベルを示し，HTML の H1, ..., H6 タグに対応して $L = 1, \ldots, 6$ が与えられる．また，末端構造に対応する $\ll desc(L) \gg$ ノードに対しては，$L = 7$ が与えられる．これらのノードは，与えられた HTML 文書から**図 2** の規則に従って導出される．以下では，特に必要がない限り，ノード種の表記では (L) を省略する．

$\ll doc \gg$ は文書全体に対応し，$\ll desc \gg$ は章や節等の構造の列に，$\ll leading \gg$ および $\ll trailing \gg$ は個々の章や節等に対応する．また，$\ll heading \gg$ は $\ll trailing \gg$ ノードの見出しに対応する．特に

$$
\begin{aligned}
\ll doc \gg &\rightarrow \texttt{<BODY>}\ \ll desc(1) \gg\ \texttt{</BODY>} \\
\ll desc(L) \gg &\rightarrow \ll leading(L) \gg \ll trailing(1) \gg + \\
&\quad \mid \ll trailing(L) \gg + \\
\ll leading(L) \gg &\rightarrow \ll block(L) \gg \\
\ll trailing(L) \gg &\rightarrow \texttt{<DIV>}\ \ll trailing(L) \gg\ \texttt{</DIV>} \\
&\quad \mid \ll packed(L) \gg \\
\ll packed(L) \gg &\rightarrow \ll heading(L) \gg \ll block(L) \gg \\
\ll block(L) \gg &\rightarrow \texttt{<DIV>}\ \ll block(L) \gg\ \texttt{</DIV>} \\
&\quad \mid \ll desc(L+1) \gg \\
\ll heading(1) \gg &\rightarrow \texttt{<H1>}text\texttt{</H1>} \\
\ll heading(2) \gg &\rightarrow \texttt{<H2>}text\texttt{</H2>} \\
\ll heading(3) \gg &\rightarrow \texttt{<H3>}text\texttt{</H3>} \\
\ll heading(4) \gg &\rightarrow \texttt{<H4>}text\texttt{</H4>} \\
\ll heading(5) \gg &\rightarrow \texttt{<H5>}text\texttt{</H5>} \\
\ll heading(6) \gg &\rightarrow \texttt{<H6>}text\texttt{</H6>} \\
\ll desc(7) \gg &\rightarrow (\ll paragraph \gg \mid \ll paradiv \gg) + \\
\ll paradiv \gg &\rightarrow \texttt{<DIV>}\ \ll paradiv \gg\ \texttt{</DIV>} \\
&\quad \mid \ll paragraph \gg \\
&\quad \mid \texttt{<BLOCKQUOTE>}\ \ll block(1) \gg \\
&\qquad\qquad \texttt{</BLOCKQUOTE>} \\
\ll paragraph \gg &\rightarrow \texttt{<P>}text\texttt{</P>} \\
&\quad \mid \texttt{<UL>}text\texttt{</UL>} \\
&\quad \mid \texttt{<OL>}text\texttt{</OL>} \\
&\quad \mid \texttt{<DL>}text\texttt{</DL>} \\
&\quad \mid \texttt{<TABLE>}text\texttt{</TABLE>} \\
&\quad \mid \text{“}text-without-structure\text{”}
\end{aligned}
$$

注 1： $\ll desc(L) \gg$, $\ll leading(L) \gg$, $\ll trailing(L) \gg$, $\ll packed(L) \gg$, $\ll block(L) \gg$ において $L = 1, 2, \ldots, 6$.

注 2： “$text-without-structure$” は任意の文字列を，“$text-without-structure$” は他の規則で受理されない文字列を受理する．すなわち，“$text-without-structure$” は BODY, DIV, BLOCKQUOTE 文書要素内の構造主体のタグを含まない文字列を受理する．

図 2 導出規則
Fig. 2 Derivation rules.

$\ll leading \gg$ は，明示的な見出しを持たない章や節に対応するが，これはそれの属する $\ll desc \gg$ の概要や後続の $\ll trailing \gg$ に対する導入部を与える傾向がある．このことは次章で利用される．また，$\ll desc(7) \gg$ は段落，リスト，表，引用文等の末端レベルの構造の列に，$\ll paradiv \gg$ および $\ll paragraph \gg$ は個々の末端レベルの構造に対応する．$\ll packed \gg$ および $\ll block \gg$ は便宜上導入した中間ノードである．

図 3 の BODY 文書要素を持つ HTML 文書からこの規則に従って導出された論理木を**図 4** に示す．論理木の末端ノードは左から順に，HTML 文書の文書要素と出現順で対応し，点線は L の値が同じノード群を表している．この例では，2 つの $\ll heading(1) \gg$ ノードが H1 に対応して導出される．また，1 章の本文に当たる左側の $\ll block(1) \gg$ 以下の点線で示される部分木においては，1 つの $\ll leading(2) \gg$ ノードが 1 章の導入の書かれた P から，2 つの $\ll heading(2) \gg$ ノー

```
<BODY>
   <H1> 1 章の見出し </H1>
   <P> 1 章の導入 </P>
   <H2> 1.1 節の見出し </H2>
   <P> 本文 1.1.1 </P>
   <P> 本文 1.1.2 <B> 重要語 </B> </P>
   <H2> 1.2 節の見出し </H2>
   <P> 本文 1.2.1 </P>
   <H1> 2 章の見出し </H1>
   <H2> 2.1 節の見出し </H2>
   <TABLE> 表 2.1.1 </TABLE>
   <P> 本文 2.1.2 </P>
</BODY>
```

図 3 サンプル HTML 文書
Fig. 3 Sample HTML document.

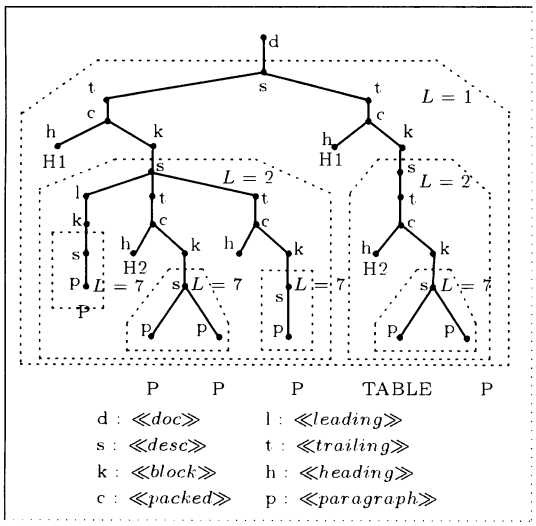

図 4 導出された論理木
Fig. 4 Derived logical tree.

ドが H2 から導出される．他のノードも同様である．

4. ビューページの生成

本章では，論理木に基づいたビューページの構築法について述べる．本手法では論理木のノードを単位としてベクトル空間モデル[7),8)]を利用するが，その特徴ベクトルの導出には 3.1 節で導入した表示主体のタグと，論理木のノードの種類を考慮する．

ビューページ生成手順の概要は次のとおりである．始めに，論理木の各ノードに対応する特徴ベクトルを求める．次に，各ノードとユーザプロファイル中のキーワード群との類似度を示すスコアを算出し，それがユーザプロファイル中に与えられる閾値より低いノードをマークする．最後に，すべての上位ノードと内部ノードがマークされている部分木を取り除くことでビューページを構築する．

表 1 表示主体のタグに与えられた重み係数 $\beta_{n,t}$
Table 1 Weights $\beta_{n,t}$ associated with style-oriented tags.

タグ	$\beta_{n,t}$	意味
STRONG	5	強い強調
EM	3	強調
BIG	3	大きな文字
U	2	下線
B	2	ボールド
I	2	イタリック
DT	2	DL での定義語

4.1 特徴ベクトルの導出

4.1.1 末端ノード

まず，全末端ノードの集合を文書集合と見なして，それらの初期特徴ベクトルを tf-idf 法[7)]に従って求める．このとき，idf 因子は末端ノードの集合に基づいて計算され，ビューページ生成対象の文書内での語の分布に従った重みを与えることになる．

次に，表示主体のタグが重要部分を示すために使われる傾向が高いことに着目する．これらのタグで囲まれた語について，あらかじめタグごとに与える重み係数を用いて重み付けを行う．この重み係数の例としては，**表 1** のようなものが考えられる．これは，重み 1 を基準とした場合に，論理的な強調を意味するものに 3 以上を，表示上の効果によって強調として使われるものに 2 を与えた例である．記載されていない表示主体のタグについては，$\beta_{n,t}=1$ であるとする．なお，ある語について，それを含むタグが複数ある場合には，それらの与える最大の重み係数を用いる．

これによって与えられる末端ノード n の特徴ベクトルは式 (1) で定式化される．ここで $\beta_{n,t}$ は，n に含まれ t が出現する文書要素の重み係数のうち，最大のものである．

$$v_n = (tf_t \cdot idf_{n,t} \cdot \beta_{n,t})_{t \in \mathcal{T}} \quad (1)$$

4.1.2 非末端ノード

非末端ノードの特徴ベクトルは，その子ノードの特徴ベクトルに基づいて，その集約として与える．

ここで，一般に見出しや概要，導入等は本文の重要語を多く含むことに着目する[8),9)]．論理木において，見出しは $\ll heading \gg$ ノードに対応する．また，概要や導入は 3.2 節で言及したように $\ll leading \gg$ ノードとして現れることが多い．このことを考慮し，非末端ノード n の特徴ベクトルは子ノードの特徴ベクトルの荷重平均を用いて次式で与える．

$$v_n = \sharp C \cdot \frac{\Sigma_c \alpha_c \cdot v_c}{\Sigma_c \alpha_c} \quad (2)$$

ここで，c は n の子ノード，$\sharp C$ は子ノードの総数であ

表 2 子ノードの重み係数 α_c
Table 2 Weights α_c for child nodes.

ノード	α_c
$\ll heading(1) \gg$	15
$\ll heading(2) \gg$	15
$\ll heading(3) \gg$	10
$\ll heading(4) \gg$	10
$\ll heading(5) \gg$	10
$\ll heading(6) \gg$	10
$\ll leading \gg$	5

る．係数 α_c は，たとえば**表 2** のようなものが考えられる．記載されていないノード種については，$\alpha_c = 1$ であるとする．特に $\ll leading \gg$ の係数の決定には，文献 9) を参考にした．

4.2 ビューページの生成

ビューページを生成するため，各ノードとユーザプロファイルの適合の度合いを決定する．まず，論理木の各ノードは，4.1 節で与えられた特徴ベクトルとユーザプロファイルとして与えられたキーワード群の特徴ベクトルとの類似度によってスコアを計算する．このとき，各ノードと対応するテキストの長さは様々である．これを考慮してスコアの算出には典型的なコサイン測度でなく，C-pivot 測度[10] を用いる（付録参照）．次に，すべての上位ノードと内部ノードに与えられたスコアがユーザプロファイルで与えられた閾値より低い部分木を取り除く．最後に，このようにして得られた論理木に対応する HTML 文書を生成する．これにより，ビューページはユーザの興味に適合する記述を含む適切な大きさの部分文書から構成されることが期待される．ただし，ビューページ生成に際しては以下の 2 点を考慮する．また，ビューページは正しい HTML 文書でなければならないことに注意を要する．

- ページ内容を理解する際には文書構造の把握が重要な役割を果たす．それゆえ，オリジナルの WWW ページに代わってビューページを提供することで文書構造の把握を妨げてはならない．特に各見出し（$\ll heading \gg$ ノードに対応）はその重要な手がかりであるから，ビューページには元のページ中のすべての見出しを含めるものとする．
- 除去された部分については，それを示す目印が与えられることが望ましい．また，単純な除去をしてしまうと期待されない結果になることがある．たとえば，“text <P> paragraph </P> text” から P 文書要素を除去した結果，“text text” となってしまい，前後の 2 つの “text” が連結されてしまう．これらから，ビューページ生成時には，除去対象の部分木が対応する文書要素列を完全に除去するのではなく，“(snip)” を内容とする DIV 文書要素で差し替えることにする．

具体的な手順は以下のようになる．

(1) 各ノードのスコアを計算する．
(2) スコアが閾値より低いノードをマークする．ただし，$\ll doc \gg$，$\ll heading \gg$ ノードは除く．これは，BODY 文書要素は必須であることと，H1,...,H6 文書要素は残すためである．
(3) 上位ノードと内部ノードがすべてマークされた極大な部分木を特定する．
(4) 特定された部分木に対応する HTML 文書要素の列を “<DIV> (snip) </DIV>” で置き換える．
(5) ビューページを出力する．

上記の手続きは，与えられた任意の正しい HTML 文書に対して，正しい HTML 文書を生成する．これは以下によって確かめられる．

図 2 に示した論理木の導出規則から，置換対象となる文書要素列は，BODY 文書要素そのものであるか，BODY，DIV，BLOCKQUOTE 文書要素のいずれかの子文書要素列である．ここで，BODY 文書要素は必須の文書要素であるが，手順 2 によって削除対象から除外される．また，BODY，DIV，BLOCKQUOTE 文書要素はいずれも DIV 文書要素を子に持つことができる．よって，上記の手続きによる置換結果は HTML の文法上，適正である．すなわち，上記の手続きは正しい HTML 文書に対して，正しい HTML 文書を生成する．

図 5 に図 3 の HTML 文書に対するビューページの生成例を示す．円は閾値以上のスコアを持つノードを示しており，手順 2 に従って，これ以外のノードから除去候補がマークされる．ここでは，点線で囲まれた部分木は置き換えの対象を示している．これから生成された HTML 文書は**図 6** である．

5. WWW 閲覧支援システム

5.1 プロトタイプシステム概要

我々は本手法を実装した，日本語に対応したプロトタイプシステムの開発を行った．日本語の処理には，形態素解析器である茶筅[11] を使用した．和文においては，形態素解析によって得られた自立語の基本形のみを語として利用し，特徴ベクトルの導出を行った．英文に対しては，各単語から語幹を抽出し，不要語を除いたものを語として利用した．実装における基本要請として，可能な限りユーザの選択肢を狭めず，現在利用されている WWW 環境をそのまま利用できることがある．本システムは次の 3 つのモジュールから構成されており，**図 7** のアーキテクチャによってこの要請を実現している．

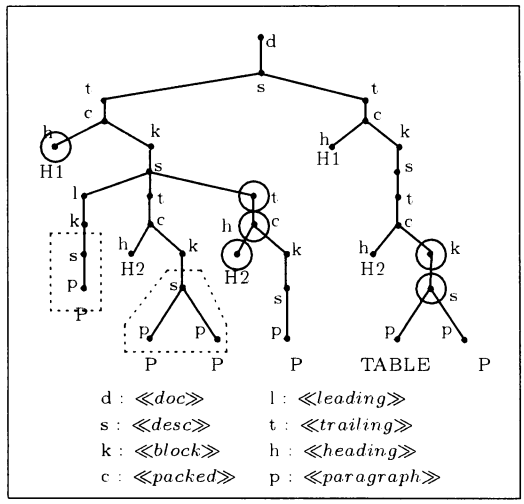

図 5 ビューページ生成例
Fig. 5 Example of view-page generation.

```
<BODY>
   <H1> 1 章の見出し </H1>
   <DIV> (snip) </DIV>
   <H2> 1.1 節の見出し </H2>
   <DIV> (snip) </DIV>
   <H2> 1.2 節の見出し </H2>
   <P> 本文 1.2.1 </P>
   <H1> 2 章の見出し </H1>
   <H2> 2.1 節の見出し </H2>
   <TABLE> 表 2.1.1 </TABLE>
   <P> 本文 2.1.2 </P>
</BODY>
```

図 6 ビューページの HTML 文書
Fig. 6 HTML document for a view-page.

- コントローラ
- ブラウザ
- 閲覧支援エンジン

実線は要求やメソッド起動の流れを、破線は応答や返値の流れを示す.

ユーザは Netscape Communicator や Internet Explorer 等の一般的に利用されている WWW ブラウザを本環境における**ブラウザ**として利用可能である. 図中の**コントローラ**, **閲覧支援エンジン**が本環境を実現するために開発したモジュールである.

ユーザはブラウザとコントローラを利用して各ページの閲覧を行う. コントローラは WWW ブラウザ上の Java アプレットとして実行され, ユーザプロファイルの入力等を担当する. また, 閲覧中のページの論理木と各ノードのスコア等の視覚的な表示機能もあわせ持つ. コントローラは, その起動時に新たなウィンドウを生成し, その中で本環境におけるブラウザとして利用する WWW ブラウザを起動する. ブラウザウィンドウに表示されたビューページでは, 通常のページと同様, リンクをたどる等の操作も可能である. また, ユーザプロファイルが変更されたときには, ビューページの内容も自動的に更新される.

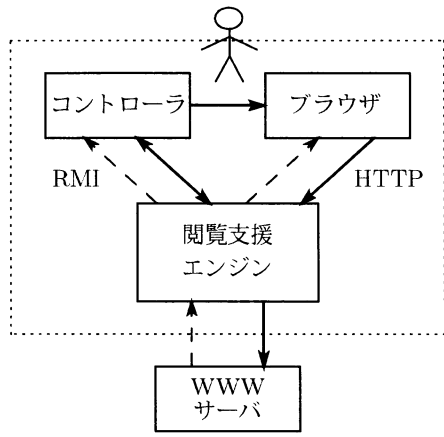

図 7 システムアーキテクチャ
Fig. 7 System architecture.

一方, 閲覧支援エンジンは独立した Java アプリケーションであり, マルチユーザ対応の WWW プロキシサーバとして動作し, ブラウザから要求されたページに対するビューページの生成・提供を行う. ビューページ生成時に参照するユーザプロファイルは, RMI を利用して各ユーザのコントローラから渡される. また, ビューページの生成は必要に応じて抑制可能である.

5.2 閲覧支援エンジン

閲覧支援エンジンの内部構機構を**図 8** に示す. これは, 主に次の 5 つのサブモジュールから構成される.

- 文書管理部
- ユーザ管理部
- HTTP 受理部
- ビュー管理部
- HTTP 処理部

初めの 3 つのサブモジュールは閲覧支援エンジンの起動時に起動される. 残りの 2 つは, 他のサブモジュールによって必要に応じて起動される. 図 8 中のユーザ管理部と HTTP 受理部から出る点線が, これらの起動とクライアントとの接続の確立を示す. 文書管理部以外のサブモジュールは, それぞれ別スレッドで実行される.

文書管理部は HTML 文書の管理を担当し, 文書のパースと DOM オブジェクト[12] の生成, 論理木の導

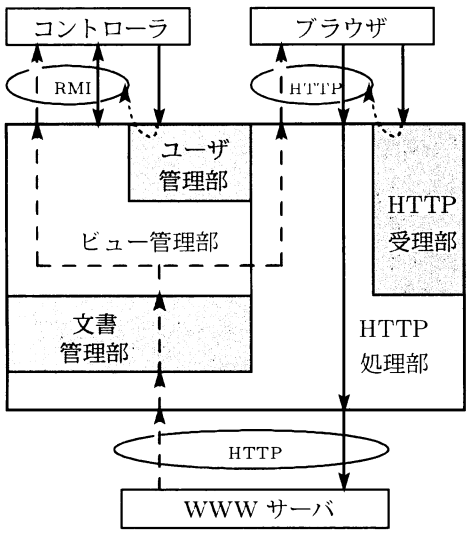

図 8 閲覧支援エンジンの内部機構
Fig. 8 Internal architecture of the Browsing Assistance Engine.

出を行う．また，それらの情報をメモリ上で，URI をキーとしたハッシュ表を用いてキャッシュする．

ユーザ管理部は，コントローラからの RMI 接続に対して待機する．各ユーザのコントローラからの要請に従い，コントローラ別にビュー管理部を生成し，それらが直接通信可能な状態に設定を行う．

ビュー管理部は，コントローラと通信してユーザプロファイルを管理する．また，それに基づくビューページの生成機能を提供する．ビューページは，文書管理部で管理される DOM オブジェクトとそれに対応する論理木から生成する．また，論理木の構造とそのノードのスコアの情報をコントローラに提供する．

HTTP 受理部は TCP ポートを監視し，HTTP 要求に応じて HTTP 処理部を別スレッドで生成する．HTTP 要求の実処理は HTTP 処理部に委譲して，ポートの監視を継続する．

HTTP 処理部は HTTP 要求を WWW サーバに転送し，リソースの情報を取得する．それが HTML 文書である場合，文書管理部にキャッシュされていなければ WWW サーバからリソースを取得し，文書管理部に登録する．さらに，ビュー管理部を利用してその文書から生成されたユーザ固有のビューページを取得し，それを HTTP 要求に対する応答として終了する．また，リソースが HTML 文書でないときにはそれを直接ブラウザに返す．

5.2.1 処理の流れ

(1) **前準備**

閲覧支援エンジンは，プロキシサーバとして利用されるため，あらかじめ起動しておく必要がある．これはコントローラアプレットを提供する WWW サーバと同じホスト上に配置する☆．まず，ユーザはコントローラアプレットを含む WWW ページへアクセスし，コントローラを起動する．コントローラは，本環境におけるブラウザが使用するためのウィンドウを生成し，以降では Java のアプレットコンテクストを通じてブラウザを制御する．さらに，閲覧支援エンジンのユーザ管理部に接続し，そのユーザ用のビュー管理部との接続を確立する．以上で，前準備は完了し，ユーザは以下の操作が可能となる．

(2) **HTTP 要求の処理**

ブラウザにおいて URI を入力したり，リンクをたどったりすることで閲覧対象の URI が指定されると，プロキシサーバとして利用される閲覧支援エンジンへブラウザから HTTP 要求が発行される．これに対し，前述の機構によって閲覧支援エンジンはそれに応じた各種リソースを応答として返す．その際，必要に応じてビューページが提供される．また，閲覧支援エンジンでは，提供したビューページの URI が前回の URI と異なっている場合にはコントローラにビューページに関する情報を提供し，ブラウザの表示と同期させる．

(3) **ユーザプロファイルの変更**

一方，ユーザはコントローラを利用してユーザプロファイルを変更できるが，その変更はコントローラからビュー管理部へ通知される．これによってビュー管理部は新しいユーザプロファイルに基づいたビューページの生成の準備ができる．さらに，この通知が完了すると，ビュー管理部から現在の URI を取得し，その URI の表示をブラウザに対して指示する．これによって HTTP 要求の処理が発生し，新しいビューページが表示される．特に，変更された内容が閾値のみであればスコアの再計算は不要である．

5.3 利 用 例

本プロトタイプシステムの利用画面例を**図 9** に示す．左側がコントローラウィンドウであり，右側がブラウザウィンドウである．ユーザプロファイルの設定はコントローラウィンドウ上部で行う．キーワード群はテキスト入力フィールドに列挙することで，閾値はスライダーを調整することで設定される．図は，ユーザプロファイルを { 閲覧支援, ブラウザ } として我々の以前の論文 5) を表示した結果である．

コントローラウィンドウでは，その下部に論理木の構造が示されており，各ノードを表す行はそのスコア

☆ この配置に関する制約は緩和可能である．

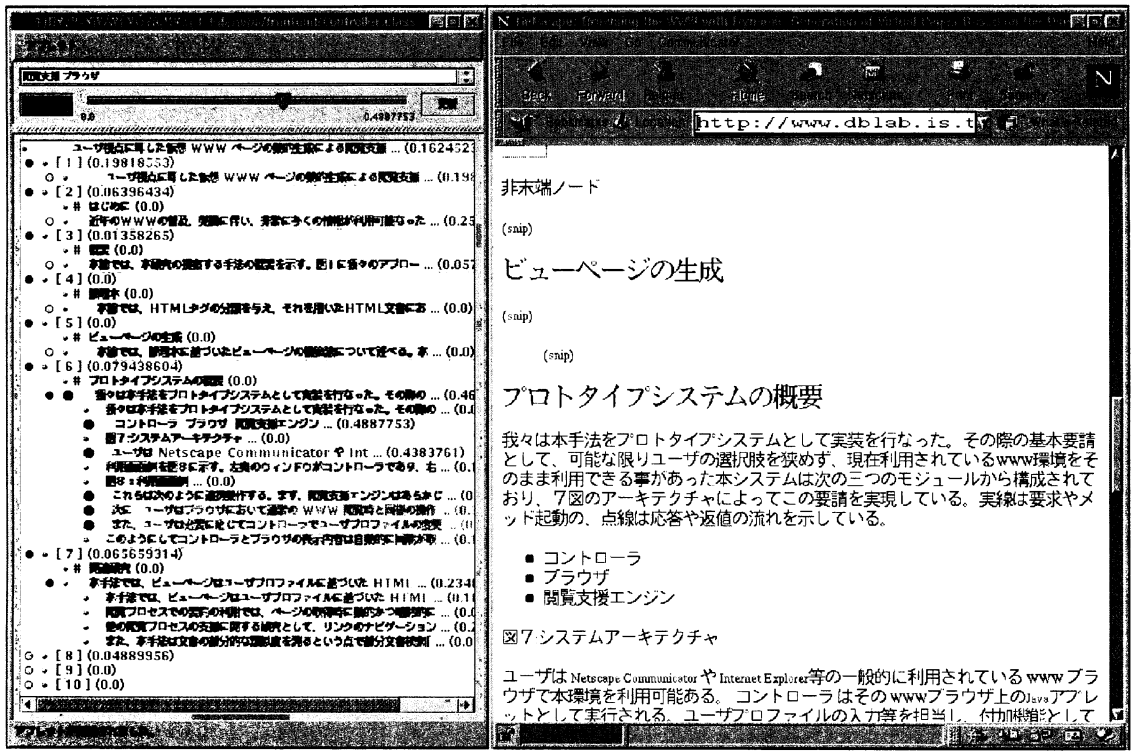

図 9 利用画面例
Fig. 9 Screen snapshot.

に応じた色で表示される．最もスコアの高いものが赤で，スコアが 0 のものが青で表示され，その中間のスコアのノードの表示色は赤～橙～黄～緑～水～青と推移する．また，特にユーザが設定した閾値以上のスコアを持つノードの左側には，大きな赤いアイコンが，それ以外のものについては小さな水色のアイコンが示される．ユーザはこれらの情報から，ユーザの欲する情報がどの部分にどの程度含まれているかを視覚的に把握でき，これを参照しながら閾値を調整できる．

一方，ブラウザウィンドウではビューページが表示されており，そのユーザにとって不要な部分の表示は抑制され，ユーザは必要な部分に着目することが容易になっている．また，通常の WWW ページと同様にリンクをたどる等の操作も可能であり，それに対しても必要に応じてビューページが提供される．

このほか，必要に応じてビューページの生成を抑制することで，検索エンジンでの検索結果はそのまま表示して，各ページの閲覧時にのみビューページを利用する，といった使い方も可能である．

6. 評価実験

本章では，提案手法の有効性を実験を通じて評価する．ここでの主な評価の対象は次の 3 点である．なお，本手法における論理木導出規則はフレームやテーブルによるレイアウトには対応していないため，そのようなページは以下では適用対象から除外する．

まず，論理木の導出について，その導出規則が妥当なものであるかを評価する．文書中の要・不要部分の選択を適切な単位で行うためには，少なくともこれが妥当なものである必要がある．

次に，論理木の各ノードのスコア付けの方法の妥当性を評価する．論理木が妥当なものであるという仮定の下でスコア付けが妥当であれば，閾値を適度に調節することによって適切な部分を抽出することができる．これによって，適切なビューページを提供できるものと考えられる．

さらに，より実際的な WWW ページ閲覧における適用評価を示す．具体的には，検索エンジン Goo にキーワード群を与え，その検索結果のページを本システムで閲覧する場合を対象に，上記と同様の妥当性の評価を行う．

以下では，それぞれの実験内容とその結果を示す．

6.1 論理木導出の妥当性

論理木導出の妥当性の評価には，次の 3 つの WWW

表 3 表示主体のタグの利用傾向
Table 3 Usage of style-oriented tags.

サイト	使用箇所	重要箇所	割合
W3C	211	153	0.725118
DLIB	341	286	0.838710
IGD	167	112	0.670659

サイトから，それぞれ 10 ページをランダムに選択して利用した．これらのサイトのページを対象とした理由は，これらが論文や技術文書等の記述量の多いページを含む比較的よく知られたサイトであり，かつ，それぞれ異なったタグ付け方針に従ったページを提供していることによる．

- http://www.igd.fhg.de/www/www95/
- http://www.w3.org/
- http://www.dlib.org/

これらのサイトでは，基本的には文章の章立てに従った見出しのタグ付けをしているが，論理木導出に利用する構成主体のタグに着目すると，たとえば次のような構成上の差異が見られる．

- 各サイトで統一性を持たせるためにロゴ等のヘッダやフッタを付加している．
- 論文のタイトルに付けられたタグと本文中の各章の見出しに付けられたタグがともに H1 であったり，前者が H1 で後者が H2 であったりする．
- サイトの方針によっては，下位の文書構造の見出しが，H5 や H6 ではなく P や B 等による表示効果を用いて記述されている．
- 参考文献リストが，DL や OL 等で記述してあったり，参考文献ごとに P で囲んであったり，サイトによってはこれらが混在していたりする．
- 特に W3C では HTML 4.0 勧告に比較的忠実であるが，一方で IGD ではオーサリングツール等によるタグの誤用やピリオドだけを内容とする無意味なタグ付け等が散見される．

また，表 1 に示したような重み付けを行うことの妥当性を調べるため，表示主体のタグが使用されている箇所について，その記述内容がビューページ生成において重要性を持っているといえるかどうかを判定してみた．その割合を算出した結果を**表 3** に示す．「使用箇所」「重要箇所」がそれぞれの箇所の数である．

上記の WWW ページに対して，章，節，段落等の最大 7 レベルの階層的文書構造を考えることとし，あらかじめ個々の文書構造の開始位置と終了位置を人手によって決定しておく．これらの境界位置の決定は，WWW ブラウザによるレンダリング結果とその記述内容を基に行った．以下では各レベルを，上位から順に h1〜h6 および block と表記する．ただし，段落，表，リスト等はつねに末端の文書構造とし，block レベルとして扱う．各レベルの文書構造の開始位置と終了位置とからなる集合を A_i $(i = h1, \ldots, h6, block)$ とする．

これを用いて，導出規則によって得られた論理木の構造の妥当性を次のように評価する．まず，論理木の各レベル L について $\ll leading(L) \gg$，$\ll trailing(L) \gg$ ノードの開始位置と終了位置とからなる集合を求め，B_i $(i = h1, \ldots, h6)$ とする．ここで，$i = h1, \ldots, h6$ は $L = 1, \ldots, 6$ に対応する．また，同様に $\ll paradiv \gg$，$\ll paragraph \gg$ ノードから B_{block} を求める．これらを用いて，各レベルの境界位置に関する適合率 P_i と再現率 R_i を次式で求める．

$$P_i = \frac{|A_i \cap B_i|}{|B_i|} \qquad R_i = \frac{|A_i \cap B_i|}{|A_i|} \qquad (3)$$
$$(i = h1, .., h6, block)$$

そのサイト別の平均結果を**表 4**〜**表 6** の左表に，全サイトの平均結果を**表 7** の左表に示す．表中の「人手」は人手で決定した境界数の合計，「本手法」は本手法の導出規則によって検出された境界数の合計である．適合率・再現率の平均を求める際，「本手法」または「人手」が 0 であったページは除外した．「件数」は適合率・再現率の平均を計算するうえで対象となったページの数である．実験結果は，特に上位レベルと末端レベルの構造において，適合率・再現率ともに高い値を示した．また，末端レベル以外の下位レベルの構造では低い値を示しているが，これらが文書中に出現するケースは全体的な傾向としては少ない．

また，各サイトにおける見出しとそれに用いるタグの関係について調べてみた．見出しタグ H1, ..., H6 を使用している箇所のうち，それが実際に h1, ..., h6 の見出しの記述を意図している箇所の割合を「見出しタグ使用傾向」として，逆に，h1, ..., h6 の各レベルの見出しのうち，それらが見出しタグ H1, ..., H6 によって記述されている箇所の割合を「見出し記述傾向」として，表 4〜表 7 の右表に示す．これより，見出しタグ使用傾向と適合率，および見出し記述傾向と再現率の相関が見て取れる．具体的には，下位レベルの見出しタグが誤用されており，本来の「見出し」としてではなく強調表示を意図して利用されていることが適合率が低下している主な要因と考えられる．逆に，下位レベルの見出しの記述を，見出しタグを使用せずにボールド表示等の表示効果を使用して記述しているページが原因となって再現率が低下しているものと思われる．

表 4 http://www.igd.fhg.de/www/www95/ に対する実験結果
Table 4 Experimental result for http://www.igd.fhg.de/www/www95/.

レベル	人手	本手法	適合率	件数	再現率	件数
h1	20	20	1.000	10	1.000	10
h2	61	60	1.000	9	0.984	9
h3	63	63	1.000	9	1.000	9
h4	19	19	1.000	2	1.000	2
h5	3	5	0.500	2	1.000	1
h6	0	0	—	0	—	0
block	549	715	0.773	10	0.998	10

見出しレベル	見出しタグ使用傾向	見出し記述傾向
h1	1.000	1.000
h2	1.000	0.990
h3	1.000	1.000
h4	1.000	1.000
h5	0.333	1.000
h6	—	—

表 5 http://www.w3.org/ に対する実験結果
Table 5 Experimental result for http://www.w3.org/.

レベル	人手	本手法	適合率	件数	再現率	件数
h1	17	17	0.850	10	0.850	10
h2	72	70	0.917	9	0.930	9
h3	68	69	0.911	9	0.933	9
h4	71	71	0.933	5	0.933	5
h5	0	0	—	0	—	0
h6	0	0	—	0	—	0
block	665	738	0.852	10	0.937	10

見出しレベル	見出しタグ使用傾向	見出し記述傾向
h1	1.000	1.000
h2	0.998	1.000
h3	0.923	1.000
h4	1.000	1.000
h5	—	—
h6	—	—

表 6 http://www.dlib.org/ に対する実験結果
Table 6 Experimental result for http://www.dlib.org/.

レベル	人手	本手法	適合率	件数	再現率	件数
h1	3	3	1.000	3	1.000	3
h2	27	27	1.000	9	1.000	9
h3	88	87	0.846	10	0.936	9
h4	46	32	0.800	5	0.667	6
h5	19	8	0.125	4	—	0
h6	0	20	0.000	7	—	0
block	659	877	0.717	10	0.947	10

見出しレベル	見出しタグ使用傾向	見出し記述傾向
h1	1.000	1.000
h2	1.000	1.000
h3	0.816	0.938
h4	1.000	0.738
h5	0.000	0.000
h6	0.000	—

表 7 全サイトに対する平均
Table 7 Averages for all sites.

レベル	人手	本手法	適合率	件数	再現率	件数
h1	40	40	0.935	23	0.935	23
h2	160	157	0.972	27	0.971	27
h3	219	219	0.916	28	0.956	27
h4	136	122	0.889	12	0.821	13
h5	24	13	0.250	6	0.667	3
h6	0	20	0.000	7	—	0
block	1873	2330	0.781	30	0.961	30

見出しレベル	見出しタグ使用傾向	見出し記述傾向
h1	1.000	1.000
h2	1.000	0.995
h3	0.879	0.940
h4	1.000	0.763
h5	0.290	0.039
h6	0.000	—

以上から，下位レベルの構造，特に H5，H6 を手がかりとした論理構造の導出に関しては，実際のページでは問題が生じる場合があることが分かった．

総合すると，本研究で対象としたような記述量の多いページについては，上位レベルと使用頻度の高い構造主体のタグに着目することで，一定の妥当性を持つ論理構造が導出できると考えられる．また，下位レベルの構造についても，その見出しが見出しタグを適切に用いて記述されている場合には，同様に比較的妥当な論理構造を導出できることが推察されるものの，実際のページにおけるタグ利用に対応した手法との組合せも今後必要と思われる．

6.2 スコア付けの妥当性

論理木の各ノードに対するスコア付けの妥当性の評価は，論理木によって論理構造が正しくモデル化できている場合を対象に行う．まず，論理構造を正しく反映している論理木を作成し，与えられたキーワード集合に対して人手による 5 段階のスコアを論理木のすべてのノードに対して与える．次に 4 章で述べた本手法によるスコアを与える．両者のスコアに基づくノードのランキング順位を比較することで，スコア付けの妥当性を評価する．

まず，対象文書として，我々の研究室の論文3編をHTML 4.0の勧告に準拠したHTML文書として編成しておく．また，各論文について2つずつユーザの視点を想定し，それぞれユーザプロファイル中のキーワード群として記述する．これらのキーワード群としては，各論文の特徴的な節ないし項を選択し，そこに集中的に頻出する5個前後の語を利用した．

次に，対象文書の論理構造を反映した論理木を人手で作成し，その中のすべてのノードについて，キーワード群に対する適合性を5段階評価で与える☆．一方，本手法によって各ノードに対してスコア付けを行う．これらの2種類のスコア付けに基づくランキング結果を，Bartellらの指標[13]に基づいて比較した．以下ではノード n について，人手によるスコアを $G(n)$，本手法によるスコアを $R(n)$ と表記する．

Bartellらの指標 J は，式(4)で与えられる．これは，ノード n, n' について全順序 $n \succ n'$ によるランキング結果 X があるとき，スコア付け関数 R によるランキング結果が X にどれだけ類似しているかを算出する．指標 J は，これらが完全に一致したときに1.0を，完全に逆順であるときに -1.0 を与える．

$$J(R) = \frac{\Sigma_{n \succ n'}(R(n) - R(n'))}{\Sigma_{n \succ n'}|R(n) - R(n')|} \qquad (4)$$

本実験では，人手で与えたスコアは5段階評価であり，半順序しか与えないため，指標 J を直接利用することができない．そこで次の2通りの定義によって基準として用いる全順序 $n \succ n'$ を与える．

(I) $G(n) > G(n') \vee (G(n) = G(n') \wedge R(n) > R(n'))$

(II) $G(n) > G(n') \vee (G(n) = G(n') \wedge R(n) < R(n'))$

人手により同じ段階の評価を与えられたノード群の順序について，定義(I)は本手法によるランキングに従うと見なす楽観的な全順序であり，定義(II)は逆順に従うと見なす悲観的な全順序である．これらの全順序について，Bartellらの指標を適用する．定義(I)，(II)に対応して指標1，2とする．

さらに本実験では，同段階のノードについてランダムの順序を与えて指標 J を計算することを100回ずつ行った．その平均をとって指標3とする．

以上の指標に基づき，本手法によるスコア付けの妥当性の評価を行い，あわせて表1，表2に示した重み係数を以下の(a)〜(c)の条件で変化させて影響を比較する．それぞれの結果は，**表8〜表10**に示す．「最大差」は最高値と最低値の差を，「向上率」は最大差と最低値との比を表す．なお，以下では $\ll leading \gg$ に対する重み係数を5に固定し，C-pivotにおける傾斜定数 S は文献10)を参考にして0.2とした☆☆．

まず，特徴ベクトルの合成における表2の影響の概要を把握するため，条件(a)として，表1を固定して表2における $\ll heading \gg$ ノードに対するすべての重み係数を一様に1，5，10，15，20とした場合について評価する．それぞれの重み付けを，all-1，all-5，

表8 条件(a)一様な重み係数 α_c を用いた実験結果
Table 8 Experiment result using uniform weights α_c.

重み付け	指標1	指標2	指標3
all-1	0.90014	0.55217	0.72589
all-5	0.90618	0.57691	0.74128
all-10	**0.90694**	0.58690	**0.74673**
all-15	0.90551	**0.58828**	0.74667
all-20	0.90201	0.58150	0.74154
最大差	0.00680	0.03473	0.02084
向上率(%)	0.75544	6.28973	2.87096

表9 条件(b)一様でない重み係数 α_c を用いた実験結果
Table 9 Experiment result using non-uniform weights α_c.

重み付け	指標1	指標2	指標3
A	0.90269	0.57477	0.73846
B	0.90501	**0.58927**	**0.74690**
C	0.90305	0.58255	0.74254
D	0.90408	0.57309	0.73832
E	**0.90590**	0.58315	0.74425
最大差	0.00321	0.01618	0.00858
向上率(%)	0.35560	2.82329	1.16210

表10 条件(c)重み係数 $\beta_{n.t}$ を変化させた実験結果
Table 10 Experimental result using different weights $\beta_{n.t}$.

重み付け	指標1	指標2	指標3
すべて1	0.90360	0.58257	0.74286
×1	**0.90501**	**0.58927**	**0.74690**
×2	0.90312	0.57783	0.74026
×3	0.90323	0.57746	0.74013
最大差	0.00189	0.01181	0.00677
向上率(%)	0.20927	2.04516	0.91470

表11 一様でない子ノードの重み係数 α_c
Table 11 Non-uniform weights α_c.

ノード	A	B	C	D	E
$\ll heading(1) \gg$	15	15	20	10	15
$\ll heading(2) \gg$	15	15	20	8	11
$\ll heading(3) \gg$	10	10	15	6	8
$\ll heading(4) \gg$	10	10	15	5	6
$\ll heading(5) \gg$	5	10	10	4	4
$\ll heading(6) \gg$	5	10	10	3	3
$\ll leading \gg$	5	5	5	5	5

☆ 値の大きなものが良い評価を与えられているものとする．

☆☆ TRECの複数のデータコレクションに対して適用したところ，それらすべてに対して効果的であったと述べられている．

..., all-20 と呼ぶ．いずれも大きな差はないが，若干ながら all-10，all-15 が良い結果を示した．ここでは平均のみを示したが，個別に比較した場合でも，ほぼ同様の傾向が見られた．また，特に all-1 はつねに最悪の結果を示したことから，重み付けが有効であると推察される．

次に，表 2 のより適切な値を探るため，条件 (b) として，同様に表 1 を固定して**表 11** に示す A, ..., E の一様でない重み付けを用いた場合について比較した．これらは，(a) の結果を考慮したうえで，主に 10〜15 程度の重み係数を割り当てた．その結果，B，E が相対的に良い結果を示し，特に B は all-10，all-15 と比べても若干の改善を示した．以上から，ノード種によって重み付けを変えることが有効であると推察される．

最後に表 1 の影響を調べるため，条件 (c) として，(b) で最善の結果を示した重み付け B について，表 1 に示した重み係数をすべて 1 とした場合，2 倍ないし 3 倍した場合についても比較を行った．その結果，表 1 をそのまま用いた結果が最も良い数値を示したが，すべて 1 としたときと比べて大きな差は見られなかった．

全体を通じて，指標 1 は全体的に十分に良いといえる結果を示した．また，指標 2 は悲観的な順序を与えたにもかかわらず，総合的には必ずしも悪いといえる結果ではなかった．指標 3 は，サンプリングによる平均的な場合の見積りとして捉えられる．これは，指標 1，2 の平均にほぼ等しい．以上の実験からは，本手法は一定の妥当性を持つスコアを与えていると考えられる．また，適切な重み付けをすることでスコア付けの改善を図ることが可能であることが示された．

6.3 実データへの適用評価

6.1 節，6.2 節では，ある程度統制されたページ群に対する，本手法の妥当性を示した．しかし，実際に公開されているページは必ずしも W3C の勧告にも従っておらず，本手法の想定外のタグ付けをしている可能性もある．そこで，本節では検索エンジンによる検索結果を閲覧するという状況を想定し，より実際的な場面での評価を行う．

まず，興味主題を決定し，それに関連する情報を多く扱っているサイトを選択する．ここでは，「アジアにおける IT 革命」に関して http://www.dir.co.jp/を，「文字コード問題」に関して http://www.horagai.com/ を選択した．これに対し，その主題のドメイン全体にかかわるキーワードと主題に特徴的なキーワードとを合計 5 つ選択し，検索キーワード群として設定する．そして，これを用いて Goo によるサイトを限定した検索を行い，上位 10 件のページを対象とする☆．これに対して，6.1 節と同様に論理木導出の適合率・再現率を求めた．さらに，表 1，表 2 による重み付けを用い，検索に使用したキーワード群をユーザプロファイルに設定して，6.2 節と同様に指標 1〜3 を求めた．

次に，サイトを限定しない検索結果に対して，同様の実験を行った．ここでのキーワード群は，上記の http://www.horagai.com/に対して用いたものをそのまま利用した．

以上の結果を，**表 12〜表 14**，および**表 15** に示す．表 12〜表 14 の表示していないレベルは，「人手」「本手法」ともに 0 であったため省略した．

論理木の適合率・再現率にはばらつきがみられたが，これは 6.1 節で用いたサイトに比べてタグの使われ方により一貫性がなかったり，本来とは異なる使われ方

表 12 http://www.dir.co.jp/ に対する実験結果
Table 12 Experimental result for http://www.dir.co.jp/.

レベル	人手	本手法	適合率	件数	再現率	件数
h1	19	17	0.958	8	0.852	9
h2	18	8	0.625	4	0.357	7
h3	26	23	0.917	3	0.688	4
h6	2	12	0.000	6	0.000	1
block	138	282	0.508	10	1.000	10

表 13 http://www.horagai.com/ に対する実験結果
Table 13 Experimental result for http://www.horagai.com/.

レベル	人手	本手法	適合率	件数	再現率	件数
h1	20	20	1.000	10	1.000	10
h2	33	33	0.808	6	0.808	6
h3	15	15	0.800	3	0.800	3
block	764	893	0.819	10	1.000	10

表 14 サイト不特定の検索結果に対する実験結果
Table 14 Experimental result for general search results.

レベル	人手	本手法	適合率	件数	再現率	件数
h1	18	16	0.875	8	0.778	9
h2	48	27	0.717	6	0.465	9
h3	30	33	0.926	5	0.833	6
h4	9	3	1.000	1	0.500	2
block	289	375	0.720	10	0.972	10

表 15 スコア付け評価指標の平均
Table 15 Evaluation of scoring scheme.

サイト	指標 1	指標 2	指標 3
dir	0.94922	0.79706	0.87235
horagai	0.88513	0.57138	0.72854
サイト不特定	0.86884	0.58719	0.72918

☆ すでに述べたように，FRAME や TABLE を使用してページ全体のレイアウトを行っているページが検索結果に含まれる場合にはそれらは除外する．

がなされていたことによる．このことから，実データへの適用においては，見出しタグによらない見出しの記述を発見したり，見出しタグの誤用を判別したりするためのヒューリスティクスを利用する等の考慮が必要であることが確認された．一方で，スコア付けの評価においては，6.2 節と比べて遜色のない結果が得られた．以上より，いくつかの課題があるものの，本手法が実データに対しても有効であると考えられる．

7. 関連研究

本手法では，ビューページをユーザプロファイルに基づいた HTML 文書の要約として生成している．文書に対する自動要約生成に関して，これまで多くの研究がなされている[7),14)]．しかし，ユーザの視点を明示的に考慮したものは少なく，ほとんどは文書内容のみに基づいて要約を生成する．

ユーザが与えた検索条件を考慮した自動要約生成手法としては Tombros らの提案がある[15)]．しかし，これは検索結果の表示を目的としており，WWW ページの HTML 文書に対するものではない．また，論理構造に基づく要約生成についても考慮されていない．

Miike らの開発した IR システム[16)] は，ユーザによる検索結果の文書の閲覧時に要約を提供することで文書内容の把握の支援を行う．これは，検索結果の文書に対してユーザの指定した詳細度に基づく要約を動的に生成し，元の文書とともにそれを表示する．しかし，その要約生成には検索条件等のユーザの視点は考慮されていない．また，WWW における閲覧支援方法についても想定されていない．

WWW における閲覧支援に関する研究の 1 つに，リンクのナビゲーション支援がある．WebWatcher[4)] はその 1 つであり，ユーザのナビゲーションパターンからユーザの興味を学習し，ページ中のどのリンクをたどるべきかを示唆するシステムである．このようなナビゲーションの支援を本手法と組み合わせることは有効であると考えられる．

また，本手法は部分文書のスコアを測るという点で部分文書検索[8),9),17),18)] と関連するが，部分文書検索では一般に閲覧支援までは考慮されていない．また，多くの研究では主に文や段落，固定長のブロック等の単純な構造を基に検索を行う．Shin らは，構造化文書の文書要素に着目した部分文書検索手法とそのためのそのための索引手法を提案している[19)]．それに対して，本手法では HTML 文書から抽出した論理構造を利用している．Willkinson は，文書検索において文書の論理構造を利用し，その部分文書の内容を考慮することで検索精度が向上することを示した[9)]．ただし，全文と節という 2 階層のみを扱っている．本研究では，より一般の多階層の文書構造を扱っている．

HTML 文書からの論理構造抽出に関して，これまでいくつかの研究が行われている．Ashish らは，WWW ページに対するラッパー自動生成の文脈において，HTML 文書からの階層的な論理構造の抽出方法について述べている[2)]．これは，本研究と同様，見出しに着目することで章や節等の階層構造を抽出する．その際，強調表示のための表示主体のタグによる記述も見出しとして扱う等のヒューリスティクスを用いるが，その妥当性の評価は与えられていない．本手法は，一般的に利用可能な構造主体のタグに基づいて論理構造の抽出を行う．また，抽出した論理構造に基づく要約生成を示すとともに，その妥当性の評価を与えた．

Embley らは，WWW ページからレコード構造を抜き出すための手法を提案した[3)]．これは，タグの出現パターンや繰返しパターン，出現頻度等に関するヒューリスティックスを組み合わせることで抽出を行う．しかし，文書の階層的論理構造の抽出は考慮されていない．

8. まとめ

本稿では，ユーザプロファイルに基づいたビューページの動的生成に基づく WWW 閲覧支援環境の実現手法を提案した．また，プロトタイプシステム実装と本手法の有効性評価実験の結果をあわせて示した．

本手法は，文書中のユーザの興味に関連する部分を特定し，WWW ページの内容を把握することを容易にする．これを行うため，HTML 文書から論理木を抽出し，ユーザプロファイルに基づいた要約をビューページとして提供する．また，プロトタイプシステム構築を通して，本手法が現在の WWW 閲覧環境の中で実装可能であることを示した．さらに，実験評価によって本手法のが一定の妥当性を持つことを確認した．

今後の研究課題としては，本手法の改善やプロトタイプシステムの改良等があげられる．特に，リンク先ページの内容を考慮したビューページ生成手法を考案する等，WWW ページ間のリンク関係を有効に活用する必要がある．また，6.1 節で述べた論理木導出における下位レベルの文書構造抽出法の改善についても検討が必要である．さらに，スコア付けの方法についても，スコア計算時の重み付けや C-pivot のパラメータの調整，ユーザプロファイルとの関係等についての，より詳細な検討を行う必要がある．≪*leading*≫ ノードが，実際にどの程度の導入部を与える傾向を持つか

を調べ，本手法への影響を調べる必要もある．本手法の XML 文書への適用も重要な課題である．

謝辞 本稿原稿に有益なコメントをくださった査読者の方に感謝する．また，本研究に関して，貴重なご意見をくださった筑波大学電子・情報工学系石川佳治講師に深く感謝する．なお，本研究の一部は文部省科学研究費補助金の助成による．

参 考 文 献

1) Atzeni, P., Mendelzon, A. and Mecca, G. (Eds.): *The World Wide Web and Databases*, LNCS, Vol.1590, Springer-Verlag (1998).
2) Ashish, N., and Knoblock, C.A.: Wrapper Generation for Semi-Structured Internet Sources, *ACM SIGMOD Records*, Vol.26, No.4, pp.8–15 (1997).
3) Embley, D.W., Jiang, Y. and Ng, Y.-K.: Record-Boundary Discovery in Web Documents, *Proc. ACM SIGMOD '99*, Philadelphia, pp.8–15 (1997).
4) Armstrong, R., Freitag, D., Joachims, T. and Mitchell, T.: WebWatcher: A Learning Apprentice for the World Wide Web, *AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments*, Stanford (1995).
5) 品川徳秀，北川博之：ユーザ視点に即した仮想 WWW ページの動的生成による閲覧支援，情報処理学会第 119 回データベースシステム研究会，Vol.99, No.61, pp.425–430 (1999).
6) Shinagawa, N. and Kitagawa, H.: Dynamic Generation and Browsing of Virtual WWW Space Based on User Profiles, *Proc. 5th International Computer Science Conference (ICSC'99)*, Hong Kong, LNCS, Vol.1749, pp.93–108, Springer-Verlag (1999).
7) Salton, G.: *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*, Addison-Wesley, Reading, MA (1989).
8) Salton, G., Allan, J. and Buckley, C.: Approaches to Passage Retrieval in Full Text Information Systems, *Proc. ACM SIGIR'93*, Pittsburgh, pp.49–58 (1993).
9) Willkinson, R.: Effective Retrieval of Structured Documents, *Proc. ACM SIGIR'94*, Dublin, pp.311–317 (1994).
10) Singhal, A., Buckley, C. and Mitra, M.: Pivoted Document Length Normalization, *Proc. ACM SIGIR'96*, Zurich, pp.21–29 (1996).
11) 松本裕治，北内 啓，山下達雄，今一 修，今村友明：日本語形態素解析システム『茶筌』version 1.0 使用説明書，奈良先端科学技術大学院大学情報科学研究科自然言語処理学講座．http://cl.aist-nara.ac.jp/lab/nlt/chasen/
12) Wood, L. (chair): *Document Object Model (DOM) Level 1 Specification.*, W3C. http://www.w3.org/TR/REC-DOM-Level-1 (1998).
13) Bartell, B.T., Cottrell, G.W. and Belew, R.K.: Optimizing Parameters in a Ranked Retrieval System Using Multi-Query Relevance Feedback, *Proc. Symposium on Document Analysis and Information Retrieval*, Las Vegas (1994).
14) Paris, C.D.: Constructing Literature Abstracts by Computer: Techniques and Prospects, *Information Processing and Management*, Vol.26, No.1, pp.171–186 (1990).
15) Tombros, A. and Sanderson, M.: Advantage of Query Biased Summarization in Information Retrieval, *Proc. ACM SIGIR'98*, Melbourne, pp.2–10 (1998).
16) Miike, S., Itoh, E., Ono, K. and Sumita, K.: A Full-Text Retrieval System with a Dynamic Abstract Generation Function, *Proc. ACM SIGIR'94*, Dublin, pp.152–161 (1994).
17) Kaszkiel, M. and Zobel, J.: Passage Retrieval Revisited, *Proc. ACM SIGIR'97*, Philadelphia, pp.21–29 (1997).
18) Zobel, J., Moffat, A., Wilkinson, R. and Sacks-Davis, R.: Efficient Retrieval of Partial Documents, *Information Processing and Management*, Vol.31, No.3, pp.361–377 (1995).
19) Shin, D., Jang, H. and Jin, H.: BUS: An Effective Indexing and Retrieval Scheme in Structured Documents, *Proc. ACM DL'98*, Pittsburgh, pp.11–28 (1998).
20) Harman, D.K.: Overview of the first TREC Text Retrieval Conference: *Proc. TREC-1*, Washington, National Institute of Standards Social Publication 500-207, pp.1–20 (1992).

付録 C-Pivot 測度

tf·idf 法を用いたベクトル空間モデルでは，文書をベクトル表現し，そのコサイン測度により文書間の類似度が測られる．しかし，単純なコサイン測度では小さな文書が比較的高い類似度を示してしまう傾向があることが知られている[20]．この問題に対し，Singhal らは文書長を考慮した **C-pivot** 測度を提案した[10]．

文書数 N，語彙集合 $\mathcal{T}$ の文書集合 $\mathcal{D}$ について，文書 d と問合せ v_q の類似度は次で与える．

$$
\begin{aligned}
sim(v_d, v_q) &= V_d \circ V_q \\
V_d &= \frac{v_d}{(1-S)\cdot L + S \cdot U_d} \\
V_q &= \frac{v_q}{\|v_q\|} \qquad (5) \\
v_x &= (tf_{x,t} \cdot idf_t)_{t \in T} \quad (x = d, q) \\
idf_t &= \log(N/n_t + 1).
\end{aligned}
$$

ここで，$d \in \mathcal{D}$ で，L は 平均語数，U_d は d 中の語の種類の数，S は $[0.1]$ の傾斜定数，$tf_{x,t}$ は語の出現回数，n_t は t を含む $\mathcal{D}$ 中の文書数である．なお，S は $\mathcal{D}$ に対して経験的に決定する必要がある．

(平成 12 年 3 月 20 日受付)
(平成 12 年 7 月 10 日採録)

(担当編集委員　角谷 和俊)

品川　徳秀（学生会員）

1973 年生．1996 年筑波大学第一学群自然学類卒業．現在，同大学大学院工学研究科博士課程在学中．構造化文書利用方式，文書データベース等に興味を持つ．ACM 学生会員．

北川　博之（正会員）

1955 年生．1978 年東京大学理学部物理学科卒業．1980 年同大学大学院理学系研究科修士課程修了．日本電気（株）勤務の後，1988 年筑波大学電子・情報工学系講師．同助教授を経て，現在，筑波大学電子・情報工学系教授．理学博士（東京大学）．異種分散情報源統合，文書データベース，構造化文書処理等に興味を持つ．著書「データベースシステム」（昭晃堂），「The Unnormalized Relational Data Model」（共著，Springer-Verlag）等．電子情報通信学会，日本ソフトウェア科学会，ACM，IEEE-CS 各会員．電子情報通信学会データ工学研究専門委員会委員長．

川田　純（学生会員）

1976 生．2000 年筑波大学第三学群情報学類卒業．現在，同大学大学院システム情報工学研究科博士課程在学中．文書データベース等に興味を持つ．