社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2008－ＦＩ－90 (5)
2008－ＤＤ－65 (5)
2008／3／28

繰り返し構造を考慮したWebページの見出しの階層構造の解析

池田彰吾† 松本章代†,‡ 小西達裕†
高木朗§ 小山照夫‖ 三宅芳雄¶ 伊東幸宏†

†静岡大学 〒432-8011 静岡県浜松市城北3-5-1
‡東京工業高等専門学校 〒193-0997 東京都八王子市椚田町1220-2
§言語情報処理研究所 〒184-0014 東京都小金井市貫井南町3-6-30
‖国立情報学研究所 〒101-8430 東京都千代田区一ツ橋2-1-2
¶中京大学 〒470-0393 愛知県豊田市貝津町床立101
E-mail: riir@inf.shizuoka.ac.jp

Webページは適切に構造化されていることが少ないため、計算機がその構造を把握するのは容易ではない．そこで本論文では繰り返し構造を発見することで，より正確にWebページ中の見出しの階層構造を解析する手法を提案する．そして，評価実験を行い，提案手法の性能を実験的に検証し，その結果を報告する．

Analysis of Hierarchy of Headlines in Web Pages by Detecting Repeated Structure

Shogo IKEDA† Akiyo MATSUMOTO†,‡ Tatsuhiro KONISHI†
Akira TAKAGI§ Teruo KOYAMA‖ Yoshio MIYAKE¶ Yukihiro ITOH†

†Shizuoka University 3-5-1 Johoku, Hamamatsu-shi, Shizuoka, 432-8011 Japan
‡Tokyo National College of Technology 1220-2,Kunugida-machi,Hachioji-shi,Tokyo,193-0997,Japan
§NLP Research Laboratory 3-6-30 Nukuiminami-cho,Koganei-shi,Tokyo,184-0014,Japan
‖National Institute of Informatics 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, 101-8430 Japan
¶Chukyo University 101 Tokodachi, Kaizu-cho, Toyota -shi, Aichi , 470-0393 Japan
E-mail: riir@inf.shizuoka.ac.jp

In this paper, we propose a method to analyze a hierarchy of headlines in Web pages by detecting repeated structure. Our method can analyze the structure of Web pages that is not well structured. We show an experimental evaluation of our method.

1. はじめに

WWW上の情報量は現在も増え続けており，それに伴い，ユーザが求める情報を素早く正確に提示する検索システムの必要性も高まっている．しかし，現在の検索エンジンは不適合ページを相当数含む結果となることが少なくなく，十分な性能とは言いがたい．検索エンジンが不適合ページを誤検出してしまう原因の一つとして，検索に用いたキーワードが全く別の文脈で独立に使用されているページであっても適合ページと判定してしまうことが挙げられる．

そこで我々は，ページ内において検索キーワードがどのような関係を持って存在しているかという点に着目して，検索エンジンの性能を向上させることを試みてきた．検索キーワードとして複数の語が用いられた場合，それらの間には何らかの意味的な関係があると考えられる．したがって，意味的関係を表現しうる構造中に検索キーワードが含まれているページを優先的に扱うことにより，検索エンジンの性能の向上が期待できる．

キーワード間の意味的関係を表現しうる構造の一つ

として，見出し構造が考えられる．見出し構造を利用した検索エンジンの性能向上については先行研究[4]で検討しているが，システムが見出し構造を正確に捉えられないことがあるという問題点があった．これは，Webページの多くが構造を表現するのに意味マークアップではなくレイアウト機能を用いていることが主な原因である．さらに，ページの制作者によって記述形式が異なることも問題を困難にしている一因である．

そこで本研究では，検索エンジンの性能を向上させるためにWebページを解析し，見出しの階層構造を抽出することを目的とする．その際，まず繰り返し構造を発見し，その後，繰り返し構造を考慮しながら見出しの階層構造を解析するという手法を提案する．

なお本研究では，現在のWWW上で最も多い，HTMLによって記述された文書を対象としている．

2. 関連研究

特にWebページの構造に着目した関連研究について述べる．

文献[3]では，繰り返し構造の検出を行うことによって同じレベルの情報のセグメンテーションを行い，Webページの構造化を行う手法を提案している．繰り返し構造に着目するという点で本研究と類似しているが，繰り返し構造のみを構造化の手がかりとしているため，明示的な繰り返し構造が存在しないページに対しては適応できないという問題がある．また，人間が構造を理解する上で大きな役割を果たしていると思われるレイアウトに関する情報を用いていないという点が本研究とは大きく異なる．

文献[2]では，テキストセグメント同士を比較し，教師あり機械学習によって親子関係を決定するという手法を提案している．階層の判断の材料には①DOMのパス②インデント情報③言語情報（先頭の記号やテキストの長さ，文末の句読点の有無，文末の品詞等）の３種類を用いている。しかしこの3種類以外にも階層構造を表現するのに用いられることが多い，視覚的な情報は利用していない．

文献[5]では，携帯電話などの画面の小さな端末にWebページを表示するためにDOM(Document Object Model)ツリーを手がかりにWebページを分割する手法を提案している．しかし，DOMはページの意味的な構造を表現しているとは限らないという問題がある．

3. 基礎的考察

3.1. 見出し構造

本研究における見出しを以下に定義する．

(ア) 一行の短い文で書かれており，他の見出しや文，図，表に対し一目で内容が分かるように付けられた標題．

(イ) 事柄をいくつかに分けて書き並べている一つ一つ．他の見出しや文，図，表などの標題にはならないものもある．箇条書き．

また，見出しの直後に存在する「見出しによって内容を表された一連の記述の範囲」を支配範囲と定義する．ある見出しの支配範囲の中に別の見出しとその支配範囲が存在し，階層構造を形成していることもある．すなわち，見出しとその支配範囲を特定するということが，見出しの階層構造を特定するということと同義である．

3.2. 繰り返し構造

図１のページにおいて矢印で示した"2004年09月13日"という見出しの支配範囲をどのように解釈すべきか．人間が目視で判断すれば"2004年09月10日"という日付の直前までであることは明らかだが，計算機にとってはこれは自明ではない．ソースファイル中には"2004年09月13日"という見出しの支配範囲を示す記述も，"2004年09月13日"が"スト騒動について〜"よりも上位の見出しであると判断できるだけの記述も無いためである．つまり，"2004年09月13日"という見出しに着目するだけでは支配範囲を正確に特定することは難しい．

ところが，このページの全体的な構造を考えると，「日付」，「見出し」，「リード文」，"続きを読む"，"posted by 〜"という並びを一塊として繰り返されて

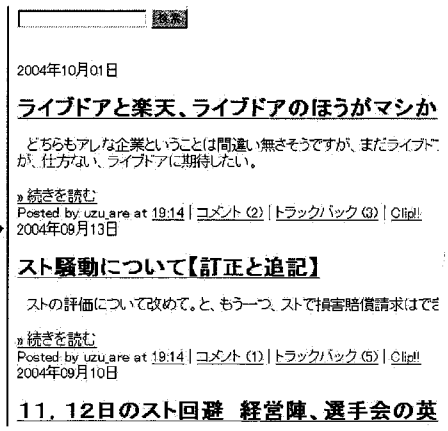

図１：繰り返し構造の一例

いることが分かる．この繰り返されているという情報があれば，"2004年09月13日"という見出しの支配範囲を判定する際に，後方に"2004年09月10日"から始まる同種の構造が繰り返されているという情報を用いて，支配範囲を適切に決定することが可能になる．このように繰り返し構造は見出しの支配範囲を特定する大きな手がかりとなる．

そこで本手法では，まず，繰り返し構造を発見することでページ全体の大まかな構造を把握する．その後，繰り返し構造を考慮しながら各見出しについて局所的な解析を行い，見出しの階層構造を特定するという２段階の手法を提案する．

なお，表はその性質上，内部に反復性を持つが，本研究では見出し構造のみを対象とするため，表の内部の解析は行わず，そこに表があるという情報だけを使用することとする．

4. 繰り返し構造の発見

本章では繰り返し構造を発見する手順について述べる．まず，見出しとその支配範囲を合わせてブロックと定義する．そして「階層構造において同レベルであり，同種の情報を持つブロックが複数隣接している構造」を繰り返し構造と定義する．つまり繰り返し構造はブロックを基本単位として繰り返されているということである．

4.1. 前処理

前処理として，解析対象であるHTMLファイルを簡単に整形しておく．すなわち省略された終了タグの補完と開始タグ・終了タグの対応関係の修正を行う．ここでは比較的簡単な処理しか行っていないが，結果的に解析の精度の向上に繋がる．

また，先行研究[1]の手法を用いて，全てのtableタグを，意味的に表を表現しているものと単にレイアウト制御に用いられているものに分類しておく．

4.2. ラベル列への変換

HTMLファイル全体をスキャンしてタグ以外のテキストを行単位で区切る．それと同時に各行の表１に示した情報（以下，属性情報と呼ぶ）を取得しておく．

表１：取得する属性情報

1	見出しタグ(h1,h2,h3,h4,h5,h6)
2	リストタグ(ul,ol,li)
3	定義リストタグ(dl,dt,dd)
4	文字の大きさ
5	強調の有無
6	リンクの有無
7	class属性
8	文字色
9	背景色
10	先頭の記号("○", "◆", "※"など)
11	先頭の連番("1.", "2-1"など)
12	括弧
13	下線の有無
14	インデント量

次に，取得した属性情報のうち，"文字の大きさ"，"強調の有無"，"リンクの有無"，"class属性"，"文字色"，"背景色"，"先頭の記号"，"先頭の連番"の８つに基づいて全ての行をいくつかのグループに分類し，全グループに重複しないようにラベル(0,1,2,3,...)を付ける．このとき同じ属性情報を持つ行同士で一つのグループを作るものとする．ただし，4.1節の前処理において表と判定されたtableタグの内部のテキストには全て"T"というラベルを付け，さらに"T"が連続している部分は一つの"T"にまとめておく．

この処理によって，Webページをラベルの列，すなわち文字列に変換したことになる．このラベル列に対して，繰り返しの発見を試みる．

4.3. 繰り返し構造の判定方法

繰り返し構造は複数ブロックによって構成されている．そのため，処理の手順としては，まず何らかの方法で繰り返し構造のブロックの候補（以下，ブロック候補）を作成し，それらが繰り返し構造を構成しうるかどうかを判定するという流れになる．

繰り返し構造の定義から，隣接していないブロック候補同士が繰り返し構造となることはないため，判定には隣接している２つのブロック候補同士を比較すればよい．このとき，比較した２つのブロック候補が全く同じラベル列であった場合は繰り返し構造であると判定すればよいことは明らかである．ところが，実際の繰り返し構造は，全く同じパタンで繰り返されているものばかりではないため，「完全一致」のみを条件とすると，繰り返し構造を取りこぼす恐れがある．そこで，ブロック候補同士の類似度を計算することで柔軟な判定を実現する．ブロック候補同士の類似度を計算する手法としてはペアワイズアラインメントを用いる．

4.4. ペアワイズアラインメント

ペアワイズアラインメントとは，入力された２つの文字列の文字間の対応関係を計算すること，もしくは，その計算結果をいう．本研究においては，比較したい２つのブロック候補を入力文字列としてペアワイズアラインメントを計算し，そのスコアを類似度として扱う．スコアがある閾値以上であれば，入力された２つのブロック候補を同じものとみなす．

ペアワイズアラインメントを用いる際に重要なのがスコア行列の導出方法である．本研究では，ページによって各ラベルが表す内容は全く異なるため，あらかじめスコア行列を用意しておくことはできない．そこで，グループ分けを行った際に用いた各行の属性情報の差異を用いてスコア行列をページごとに動的に生成することにする．すなわち，各ラベル同士で異なっている属性情報の数に応じたペナルティによってスコア行列を構成する．ただし，属性情報の中でも比較的重要なものとそうでないものがあると考えられるため，あらかじめ属性の重要度を調査した．調査の手順を以下に示す．

I. 4.2.節の処理によってグループ分けされた状態Aと，人手で理想的にグループ分けした状態Bを作成する．

II. Bでは同じグループに分類されているが，Aでは別々のグループに分類されている行の対を全て抽出する．

III. 一組の行の対を取り出し，２つの行の間の属性情報の差異，すなわちどの属性情報の違いを無視すれば同一グループに分類できたかを調べる．

IV. 全ての行の対について同様にして調べ，各属性情報ごとに異なっている数を数える．

こうして得られた値の繰り返し構造の個数に対する割合の逆数を各属性が異なっていた場合のペナルティとする（表２）．ただし，８つの属性情報のうち文字の大きさだけは唯一，一致・不一致という二値ではなく値の差を考えることができるので，ペナルティと差の積を用いるものとする．そして，各行ごとに属性情報を比較し，異なっている属性情報に応じたペナルティの合計値によって，スコア行列を生成する．

また，ギャップスコアとブロック候補同士を同一視するかどうかの閾値については，６章で詳述するテストコレクションを対象とした実験結果との調整により，それぞれ-22.0，-90.0とした．

表２：各属性情報のペナルティ

文字の大きさ	-24.55	文字色	-11.25
強調の有無	-19.29	背景色	-12.85
リンクの有無	-5.13	先頭の記号	-20.77
class属性	-9.64	先頭の連番	-29.41

4.5. 繰り返し構造の探索

以下の手順で繰り返し構造の候補を探索する．

① ラベル列をS，S内において２回以上出現[*1]する，"T"以外のラベル[*2]の集合を$L=\{L_1,L_2,L_3,\ldots L_n\}$，

[*1] ２回以上出現しないならば，そのラベルで繰り返し構造を構成することはありえないため．

[*2] 繰り返し構造の定義により，表を先頭とするブロックによって繰り返されることはありえないため．

繰り返し構造を構成するブロックの集合をR={}，S内のラベルを指すポインタをiとする．

② Lからあるラベルを１つ取り出す．取り出されたラベルをL_xとする．

③ ポインタiをSの先頭にセットする．

④ iを１ずつ後方へずらしてゆき，L_xが出現した位置をブロック候補の先頭とする．

⑤ iを１ずつ後方へずらしてゆき，「L_xが次に出現する箇所の直前」か「Rに含まれるブロックの先頭の直前」か「Rに含まれるブロックの末尾」までをブロック候補の末尾とする．ただし，ラベル１個分の長さしか持たないものは破棄する．

⑥ iを「ブロック候補の末尾＋１」にセットする．

⑦ ④～⑥の処理をiがSの末尾に辿り着くまで繰り返し，ブロック候補群を作成する．

⑧ iがSの末尾に辿り着いたら，最終ブロック候補の作成（次節にて詳述）を行う．

⑨ ②～⑧で作成されたブロック候補群に対して繰り返し構造を構成しうるかどうかの判定を行う．

⑩ ⑨で繰り返し構造であると判定されたブロック候補群をRに追加する．ただし，Rに追加すべきブロックが無かった場合は終了する(*)．

⑪ ②～⑩の処理を，Lが空になるまで繰り返す．

このアルゴリズムでは，Lからラベルを取り出す順序によって全く異なる繰り返し構造が出力されるため，全ての順序の組合せについて，①～⑪の手順を実行する必要がある．ラベルの種類数はnなのでラベルを取り出す順序の組合せの数はn!であり，最大でn!回，①～⑪のループを繰り返すことになる．しかし実際には，ほとんどの場合(*)の条件によって処理が終了するため，Lが空になるまでループが繰り返されることはまずない．n!種類の各組み合わせに対して，繰り返し構造を構成できるラベルの種類数を平均でmとすると実際の実行回数は${}_nP_m$にまで減少するため，計算量は十分計算可能な範囲に収まる．

そして，システムは生成された最大n!種類の繰り返し構造の候補の中から，繰り返し回数の和が最大である構造を最適な構造として選択する．繰り返し回数の和とは，繰り返し構造と判定されたブロックの個数の構造全体での合計値である．

4.6. 最終ブロック候補の作成

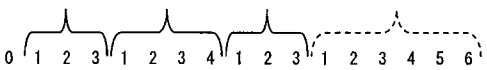

図２：最終ブロックの末尾

前節のアルゴリズムの⑧における最終ブロック候補の作成について述べる．図２のラベル列は，"1,2,3"や"1,2,3,4"など"1"を先頭とするブロックによって繰り返されていると考えることができるが，一番後ろのブロックだけは末尾が自明ではない．そこで，最終ブロック候補だけはラベルを１つずつ延長しながら既に確定しているブロック候補とペアワイズアラインメントによりスコアを計算してゆく．そしてスコアが最大の箇所を最終ブロック候補の末尾とする．ただし，スコアが閾値を超えなかった場合は破棄する．

5. 見出しの階層構造の解析

本章では，4章の処理によって発見された繰り返し構造を用いながら，実際に見出しの階層構造を解析する手法について述べる．

5.1. 見出しの検出

Webページ中にはさまざまな方法によって表現されている見出しが存在する．そこで以下の３種類の方法によって表現されているテキストを見出しとして検出する．

A. 見出しを表すタグを用いて表されているもの

具体的には，見出しタグ(h1～h6)，リストタグ(ul,ol,li)，用語定義タグ(dl,dd,dt)などを用いて表されているものである．

B. レイアウト機能を用いて表されているもの

具体的には，以下のようなものである．

- 文頭に数字（"1-1"など）や文字（"a."など）が連番で存在するか，記号（"●"や"□"など）が存在する．
- 行全体が強調タグによって強調されている．

・ 行全体にデフォルト色以外の色が指定されている．
・ 行全体がリンクとなっている．
・ 一つのセル内のテキストの末尾にコロン":"が存在している．

C. クラス名から推定できるもの

タグのclass属性が指定されており，その値に"midashi"や"title"を含む．

こうして検出された見出しに対して，さらに繰り返し構造を考慮した修正を加える．繰り返し構造を構成するブロックの先頭行のうち，１つでも見出しと判定されている場合は，他のブロックの先頭行も見出しとして検出する．

5.2. 見出しの階層構造の判定

前節までで検出された見出しを用いて，見出しの階層構造判定を行う手法について述べる．

ある見出しAの支配範囲の終了箇所はAと同レベル以上の見出しが出現する直前までと考えられる．そこで，AとAの後方に出現する見出しを順次比較してゆき，その親子関係を決定することでAの支配範囲を決定することができる．しかし，見出しの比較を行う前に，繰り返し構造に関する情報があれば，見出しの支配範囲はある程度絞り込める．そこで，繰り返し構造と見出しの支配範囲の関係について考察する．

・ Aを含む繰り返し構造とAの支配範囲の関係

Aを含む繰り返し構造が存在するならば，Aの支配範囲の終了箇所は最長でもAを含むブロックの末尾までである．なぜならAを含むブロックの末尾を越えると，支配範囲が交差することになってしまうからである．また，特にAがブロックの先頭である場合はAの支配範囲の終了箇所はAを含むブロックの末尾である．

・ Aを含まない繰り返し構造とAの支配範囲の関係

Aの支配範囲の終了箇所がAの後方に存在する繰り返し構造の途中になることはない．一つの繰り返し構造を構成するブロックは全て，同一の見出しの支配範囲内になければならないからである．

このようにして見出しの支配範囲の終了箇所をある程度絞りこんでから，２つの見出し同士を比較し，親子関係を決定する．見出しの親子関係の判断材料としては，(1)文字の大きさ(2)文頭の記号の有無(3)背景色(4)強調の有無(5)下線の有無(6)インデント量の６つを用意し，ヒューリスティックに基づいて親子関係を決定するアルゴリズムを構築した（図３）．

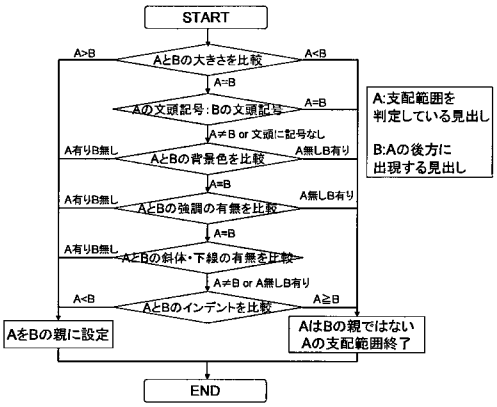

図３：親子関係判定アルゴリズム

6. 評価実験

本章では4と5章で述べた手法の性能の評価について述べる．

6.1. 評価用テストコレクション

評価用テストコレクションの作成手順を以下に示す．

1 200ページの選定．

1.1 ２語のキーワード対118組[*3]を用意しその118組をGoogleで検索し，検索結果上位から10ページずつ合計1180ページ用意する．

1.2 サイズが0バイトのページを削除する．

1.3 残りをサイズ順にソートする．

1.4 1.3の結果から中間の順位200ページを仮候補とする．

1.5 ドメインが同じページを削除し，減少した分を新たに追加する．

1.6 キーワード対の偏りをなくすために，キーワード対ごとに最大で３ページとなるよ

[*3] Webページ 10 万ページを取得し形態素解析を行い，TF/IDFで上位となった語から自然な組合せを選ぶという手法で決めた．

うに，ページの追加・削除を行う．

2　繰り返し構造，見出し，支配範囲情報の付与．

2.1　被験者（大学生３名）に見出しと支配範囲を判定してもらい，３人の判断が一致した170ページをテストセットとする．

2.2　被験者（大学生３名）に，繰り返し構造を判定してもらい，３人の判断が一致したもののみを繰り返し構造として扱う．

2.3　170ページに対して見出し，支配範囲，繰り返し構造に関する情報を付与する．

2.4　クローズドテストセット50ページ，オープンテストセット120ページとし，これらを評価用テストコレクションとする．

なお，4章，5章で述べた手法はクローズドテストセットのみを分析に用いて構築した．

6.2. 繰り返し構造の発見手法の評価

前節の手順で作成したテストコレクションを用いて，4章で述べた繰り返し構造判定手法の性能の評価を行った．評価するにあたり，以下の３種類の一致の基準を用意した．

- 完全一致：正解の繰り返し構造とシステムが出力した繰り返し構造が全て一致する．
- 部分一致：正解の繰り返し構造とシステムが出力した繰り返し構造に共通部分が存在する．
- 不一致：正解の繰り返し構造とシステムが出力した繰り返し構造に共通部分が存在しない．

結果を表３に示す．

表３：繰り返し構造判定の評価結果

	perfect	partial	fa	miss	precision	recall
closed	98	70	49	116	0.452	0.345
open	199	161	135	301	0.402	0.301

closedはクローズドテスト，openはオープンテストを表す．また，perfectは完全一致の個数，partialは部分一致の個数，faは誤検出による不一致，missは取りこぼしによる不一致である．precisionとrecallはそれぞれ以下の式で計算される．

precision = perfect / (perfect + partial + fa)

recall = perfect / (perfect + partial + miss)

両テストセット共に50%に満たないprecision, recallであり，十分な性能であるとは言いがたい．そこで繰り返し構造の発見に失敗したページを調査したところ大きく二つの原因があることが分かった．

一つは，4.3節で述べたブロック候補同士の比較において繰り返し構造の判定に失敗したというケースである．特に上位階層のブロックほど長くなるためにスコアが悪くなりやすく，その結果判定に失敗することが多い．ブロック候補を作成するところまでは正しく処理できているため，部分一致になりやすい．表３において部分一致の数が多い主な原因である．この問題はペアワイズアラインメントのペナルティの調整や閾値の調整などを行うことにより，改善できる可能性がある．また，ブロック同士の包含関係から上位階層のブロックを判定し，上位階層のブロック同士の比較においては閾値を下げるなどの手法も考えられる．

もう一つの原因は繰り返し構造の評価方法の問題である．4.5節において繰り返し構造の探索が終了した後，繰り返し回数の和が最大となる構造を選択すると述べた．しかし分析の結果，最良の構造が繰り返し回数の和が必ずしも最大になるとは限らない，ということが分かった．最良の構造とは多くの場合，バランスよく階層化されている構造である．しかし，繰り返し回数の和という点で見ると，フラットな構造の方が多くなってしまう場合がある．よって，単純に繰り返し回数を加算するのではなく，階層のレベルに応じた重み付け等を現在検討中である．

6.3. 見出し検出手法の評価

5.1.節で述べた見出し検出手法の性能の評価を行った．結果を表４に示す．

表４：見出し検出手法の評価結果[*4]

	hit	fa	miss	precision	recall
closed	1381	531	573	0.722	0.707
open	2659	1226	1207	0.684	0.688

[*4] hit：プログラムで正解を検出できたもの
fa：プログラムで不正解を誤検出したもの
miss：プログラムで正解を取りこぼしたもの
precision=hit/(hit+fa)　recall=hit/(hit+miss)

十分高精度とまでは言えないが，ある程度の見出しを検出することには成功している．

fa の原因を調査したところ，一行全体がリンクであるものを見出しと判定するという手法での誤検出が多かった．また miss に関しては，テキストの意味の理解を行う必要があるものが多かった．

6.4. 見出しの階層構造判定手法の評価

5.2.節で述べた見出しの階層構造判定手法の性能の評価を行った．まず，評価用テストコレクションから見出しをノードとした木構造を出力する．次に5.2.節で述べた見出しの階層構造判定手法を実行した結果得られる木構造を出力する．この二つの木構造から，先祖子孫関係にある全ての見出し対を抽出し，比較する．結果を表５に示す．なお，この比較において先祖子孫間の距離は考慮していない

表５：先祖子孫関係判定の評価結果*4

	hit	fa	miss	precision	recall
closed	2598	1088	1513	0.705	0.632
open	3876	2331	3131	0.624	0.553

繰り返し構造を考慮したことで支配範囲が変化した見出しは全部で421個あり，そのうち支配範囲を取り誤ってしまったものは172個であった．それらの見出しを目視で分析したところ，156個(約91%) は繰り返し構造の取り誤りが原因だった．よって，繰り返し構造を正しく検出することで，さらに精度を向上させることができると考えられる．

7. 考察

本論文で述べた手法では，繰り返し構造を検出してから，見出し・支配範囲の判定を行う．しかし，繰り返し構造の発見よりも，見出しの検出を先に行うという手法も考えられる．あらかじめ，ある程度見出しを発見しておけば，それらの見出しに特に着目しながら繰り返し構造を探索することができる．これによって，精度の向上だけでなく，計算量の減少も期待できる．

また，本研究ではタグを構造を表現するものとしては扱うことはあまりせず，Webページの見た目という点を重視して構造解析を試みてきた．しかし，タグのパス情報などは繰り返し構造を発見する上で重要な手がかりになりうる。例えばかかっているタグが全て同一であるテキスト同士は繰り返し構造となる可能性が高いと考えられる。このようなタグ情報の利用方法についても今後，詳細な検討が必要である。

8. おわりに

本論文では，WWW検索エンジンの検索精度を向上させるために，Webページの見出しの階層構造を解析する手法について議論した．そして，繰り返し構造の発見と見出し・支配範囲判定という２段階の処理を組み合せたシステムを提案・構築した．

今後の課題としては，6.2.節で述べた点を見直し，特に繰り返し構造発見の精度を向上させる必要があると考えている．そして，本論文で提案したシステム全体を先行研究の検索システムに組み込み，検索性能の評価を行う予定である．

参考文献

[1] 松本章代，小西達裕，高木朗，小山照夫，三宅芳雄，伊東幸宏：表構造における意味的関係に基づくWWW検索性能の向上，電子情報通信学会論文誌，D Vol.J91-D No.3 (2008)

[2] 松本吉司，高橋哲朗，乾健太郎，松本裕治：Webページのテキストセグメント階層構造の抽出，言語処理学会，大11回年次大会，発表論文集，vol.11th,pp.49-52,2005

[3] 南野朋之，齋藤豪，奥村学：繰返し構造に基づいたWebページの構造化，情報処理学会論文誌，vol.49, No.9, pp.2157-2167(2004)

[4] 西口直樹，松本章代，小西達裕，高木朗，小山照夫，三宅芳雄，伊東幸宏：見出しの階層関係を利用したWWW検索精度の改善，電子情報通信学会技術研究報告，NLC2005-114, pp.1-6(2006)

[5] Yu Chen,Wei-Ying Ma,Hong-Jiang Zhang : Detecting Web Page Structure for Adaptive Viewing on Small Form Factor Divices.In Proc. World Wide Web Conference 2003,2003.