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

2004－ＦＩ－74　(10)
2004－ＤＤ－43　(10)
2004／3／26

HTML 構造における頻出パターンのマイニングによる WWW からの情報抽出

清水　力　　相田　仁

東京大学大学院 新領域創成科学研究科
E-mail:　{shimizu, aida}@aida.k.u-tokyo.ac.jp

あらまし　WWW の急激な普及に伴い, 大量の HTML 文書が WWW 上に蓄積されている. 蓄積された文書は様々な用途に期待されている一方で, そのほとんどが半構造化データである HTML 文書として製作されているために機械的に扱うことが難しいという問題がある. 近年 HTML 文書における頻出パターンの抽出にもとづいたデータマイニングが注目されているが, その一環として本論文では HTML 文書から効果的に情報を抽出するために解析対象の文書を起点に同一 Web サイト内の他の文書を収集し，解析の参考とする手法を提案する．同一 Web サイト内における各 HTML 文書のタグに関する tf/idf 値から文書間の類似度を算出し，類似する文書間のパターン解析をすることで情報抽出の精度を高めることを目指し，提案した手法の評価を行った.

Information Extraction from the WWW by Mining Frequent Graph Patterns in HTML Structures

Chikara SHIMIZU　　Hitoshi AIDA

School of Frontier Sciences, University of Tokyo
E-mail:　{shimizu, aida}@aida.k.u-tokyo.ac.jp

Abstract　The scale of the WWW is growing steadily. The WWW is now expected to be a valuable information source, but are difficult to handle automatically, since almost all of existing contents are written in semi-structured language, such as HTML and XML. Information extraction methods in such environment, by mining frequent graph patterns in semi-structured documents are studied widely. We propose a method to improve the availability, by collecting documents found in the same web site, and analyzing similar documents, and present the evaluation of this method.

1. まえがき

1.1. 情報源としての WWW

WWW の急激な普及に伴い，昨今では大量の HTML 文書が蓄積されつつある．WWW の規模が増大に伴って有用なデータの絶対量が確実に増えている一方で大規模化による処理コストの増加や提供されるデータの多様化によって必要な情報を正確に取り出すことは難しくなってきている.

近年では WWW の有用性を前提にした上で幅広い分野の専門家が，この問題にさまざまな方面から取り組んでいる．WWW における登場人物は大きく分けて二人いる．コンテンツの供給者と利用者である.

Semantic Web はその中の重要な取り組みの一つであり近年その言葉自体広く認識されてきているが，この観点では供給者側環境を改善することを目指した技術であるといえる．Semantic Web が描く将来像においては WWW 上には機械処理が容易な構造化データが整備され，例えばその環境においては利用者が放ったモバイルマルチエージェントプログラムが利用者の命令を忠実に遂行し，利用者の日常生活を支援するということが容易に実現できる.

もう一方の利用者側立場にたった技術として Web Data Mining[1]がある．Data Mining とはもともと膨大なデータソースの中から有用な情報を抽出するための技術だが，その対象の一つとして WWW が取り上げられることが多くなってきている．Web Data Mining は WWW が有するどのデータに着目したものかによって手法も成果も変わってくるため一概にまとめることはできないが，現在幅広く研究されている Web Data Mining の技術が進歩すれば必要な情報と無用な情報を自由自在にフィルタリングして，蓄積された膨大な情報を無駄なく活用することができる．Web Data Mining の研究分野を表 1に示す.

どちらの技術も実現すれば豊かな情報ソースの活用を支援する有用なツールの実現につながるが，どちらも抱えている課題は多いというのが現状である.

表1　Web Data Mining の分類

	Web Mining			
	Web Content Mining		Web Structure Mining	Web Usage Mining
	IR View	DB View		
View of Data	Unstructured Semi structured	Semi structured Web site as DB	Links structure	Interactivity
Main Data	Text documents Hypertext documents	Hypertext documents	Links structure	Server logs Browser logs
Representation	Bag of words, n-grams Terms, phrases Concepts or ontology Relational	Edge-labeled graph (OEM) Relational	Graph	Relational table Graph
Method	TFIDF and variants Machine learning Statistical (including NLP)	Proprietary algorithms ILP (Modified) association rules	Proprietary algorithms	Machine learning Statistical (Modified) association rules
Application Categories	Categorization Clustering Finding extraction rules Finding patterns in text User modeling	Finding frequent substructures Web site schema discovery	Categorization Clustering	Site construction, adaptation, and management Marketing User modeling

1.2. 研究の目的

利用者の情報獲得行動をより幅広く支援するようなシステムの需要は高い.

現在利用者にキーワードを指定させ，そのキーワードが含まれる Web ページを検索するタイプの検索エンジンが広く利用されている．これらの検索エンジンは現時点では有効なツールであるが，文字列のマッチングしか行わないためにさまざまな不都合が生じる.

例えば「中野にある歯科医」を調べたい場合，「中野」「歯科医」というキーワードで検索エンジンに問い合わせると，「中野にある歯科医」と「中野さんの歯科医」が同時に取得される．このような間違いを回避するためには，前者の「中野」が「住所」という属性の一部であり，後者の「中野」が「氏名」という属性の一部であることを処理の中で加味する必要がある.

このように文字列のマッチングに加えて意味の解釈まで踏み込むことで，例えば「17 インチ以上の液晶モニタ，512MB 以上のメモリ，80GB 以上の HDD を搭載した 20 万円以下の PC」というような複雑な条件をユーザーに指定させ，その回答を与えることができるようなシステムが実現できる．そのようなシステムの構想を図 1に示す.

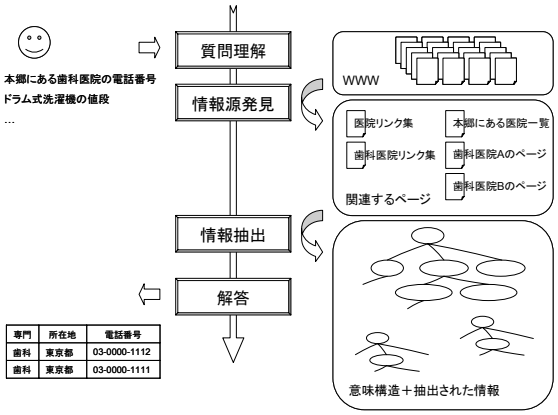


図1　WWW からの情報抽出

情報獲得の対象は，WWW 上に最も多く存在している HTML 文書とする．HTML 文書から情報を獲得することの本質的な難しさは，HTML 文書が半構造化データであることに起因する．構造的な性質とそうでない性質を併せ持つ HTML 文書に対して，構造的な性質に焦点を絞った解析手法を以降の章で紹介していく.

1.3. Semantic Web

先述した Semantic Web という概念のもとで進められている機械処理可能な仕様の標準化作業，さらにその標準に則したエディタやブラウザの整備が完了した場合，果たして従来研究されてきた情報抽出技術は利用されなくなってしまうのだろうか.

Semantic Web 技術の実用化は遅かれ早かれ，近いうちに実現する．そして重要なコンテンツ，言い換えるとコストをかけて作成することが許されるコンテンツに関しては積極的にその仕様に即した製作が推奨されるだろう．一方で現在 WWW 上では個人の日記のような，製作コストの許容値が低いコンテンツも多く存在している．そのように許容が低いコンテンツに関しては，従来の HTML を用いた製作が完全に置き換わることはないだろう.

そのため，例え Semantic Web の時代が到来したとしても従来行われている HTML 文書からの情報抽出に関する研究の成果は，依然必要になってくるだろう.

1.4. 自然言語処理

既存の研究として HTML 文書をはじめさまざまなテキストコーパスに対して言語処理的手法を用いた情報抽出に関する研究は多く行われている．その例として形態素の解析などをはじめとするアプローチなどがあるが，それらは HTML 文書におけるタグ木構造を解析するアプローチとは独立な関係にあると考えられる．つまり，両者は相互補完的な関係にあり，それぞれで有効な技術が開発された場合それらを組み合わせることでさらにその性能が改善される関係にある．その上で我々は言語処理的な手法には触れず，木構造の解析だけでどこまで情報抽出が可能なのかを追求している.

2. HTML 文書からの情報抽出

2.1. 構造化データと半構造化データ

WWW において提供されるコンテンツの中で圧倒的に多いのが HTML 文書である. HTML は WWW に関する技術の標準化団体である W3C によって規定された，文書をマークアップするための仕様である.

HTML 文書はテキストデータ，それを階層的に分断する HTML タグと呼ばれるタグ付け，そして外部コンテンツへの参照を基本としたハイパーテキストである.

HTML タグセットの中には表(<TABLE>)やリスト(<OL>)などデータ構造とデザインの両方を定義するようなタグも多いが，文書内データの構造化よりは純粋な見た目(デザイン)に関心が払われることが多く，WWW 上に存在するほとんど全ての HTML 文書は半構造化データであるということができる．以下に文書の構造化について説明をした上で，HTML 文書の特徴についてさらに踏み込んで述べる.

2.2. 構造化データ

構造化文書とは，構造の厳格なモデルを持ち，そのモデルに忠実に沿って記述された文書のことである．モデルが持つ厳格さの線引きは難しく，厳密にどこからが構造化文書で，というようにはっきりと境界を定めることはできない.

例えば XML において,「ある HTML 文書の著者が A 氏である」ことを記述するためには，表 2から表 4に示すように何通りかの表現方法が存在する.

表2　XML 表記の例

```
<文書 所在地="URL" 著者=" A 氏" />
```

表3　XML 表記の例

```
<文書 所在地="URL">
  <著者>A 氏</著者>
</文書>
```

表4　XML 表記の例

```
<著者>
  <文書>URL</文書>
  <氏名>A 氏</氏名>
</著者>
```

しかし構造の定義の仕方にこのような冗長性があると，構造化された文書を機械処理する際に構造と意味の対応関係を解析する必要が生じるため，構造化のメリットが相殺されてしまう.

そこで，構造化のモデルをより厳密に定義するための手段として DTD(Document Type Definition)， XML Schema, RELAX(REgular Language description for XML)などが存在する.

DTD を用いた場合，表 5のように定義することで上記の冗長性を排除することが出来る．このモデルにおいては「文書」要素が要素を持たず,「所在地」と「著者」という二つの属性を持つことが定義されているため, 表 2に示された XML 表記だけが正しい表現として認められる.

表5　DTD 表記の例

```
<!ELEMENT 文書 ()>
<!ATTLIST 文書 所在地 CDATA>
<!ATTLIST 文書 著者 CDATA>
```

また DTD にはいくつか実用上機能的に足りない部分があったために，RELAX や XML Schema においてさまざまな改良がなされている．RELAX は DTD に対して次の点において優れている.

- データ定義構文に XML 構文を採用している
- 豊富なデータ型を備えている
- 名前空間を扱うことができる

XML Schema は DTD に対して大幅な拡張を盛り込んだ内容になっており機能は RELAX よりも多いが, DTD の範囲を大きく超えている上にさまざまな業界団体が策定にかかわったため，難解な仕様として完成してしまったという経緯がある.

2.3. 半構造化データ

半構造化データ[2]とは生のデータ(Raw Data)でなく，また厳格に構造化されたデータ(Structured Data)でもない．半構造化データは次のように説明することができる.

- 構造が不規則なデータの集まりで，はっきりとしたスキーマが定義できない
- スキーマはあるが非常に緩やかで，全てのデータが厳密にスキーマに適合しているとは限らない
- データベース以外の形で蓄積されており，データベースのようにあらかじめスキーマが提示されているわけではない
- データ情報の中にスキーマ情報が存在する
- 構造を司るルールが不規則であったり，構造自体の適用が局所的だったりする

HTML 文書は, HTML で記述されたタグ付きの構造化データであるが，個々のタグがタグの入れ子構造において，どの深さのレベルに出現するかがあらかじめ定義されていない．そのため，HTML 文書は半構造化データとみなすことができる．同様な半構造化データの例としてはハイパーテキスト, XML, SGML, BibTeX(文献情報)などがある.

半構造化データである HTML の特徴としては，次のものがある.

- タグに囲まれたテキスト部分の型が明確に定まっていない
- 同じ意味構造を多様な HTML 表記で記述することができ，さらにその変換を行うための情報を持たない

半構造化データは上述のように，非常に曖昧な定義

づけのもとに利用されている概念であるが，ここでは位置づけをはっきりさせるために，半構造化データを次の様に分けて考える．

- 構造的な要素が意味の構造と強く対応している場合
- 構造的な要素と意味の構造が必ずしも対応していない場合

本研究においては後者を対象とする．前者の例としてはBibTeX,後者の例としてはHTMLが挙げられる．

2.4. 半構造化データの研究事例

2.4.1. BibTeXからの情報抽出

丸山ら[4]はBibTeX形式のデータを対象として，スキーマパターンを抽出した上でそれにもとづいて相関ルール抽出を行うという手法を提案している．先述したように同じ半構造化データの中に分類されつつも，HTMLとBibTeXでは後者のほうがタグ構造と意味構造の対応関係を汲み取りやすく，HTMLを対象とした情報抽出の方が難しいといえる．

2.4.2. 属性名・値の抽出

半構造化データから構造化された情報を抽出する際，もっとも重要になるのが属性名と属性値の対応関係を抽出する問題である．

Wangらの半構造化データにおけるSchema Discoveryに関する研究[5]においては，属性名と属性値はそれぞれ次のように与えられることとしている．

属性名

- タグで囲まれた文字列

属性値

- <A>タグで囲まれた文字列，通常の文字列

塚本ら[6]はHTML文書における<TABLE>構造の認識を行う際に，テーブル内各セルの特徴ベクトルからセル間の類似度，行間の類似度，列間の類似度を求めた上で行間もしくは列間において類似度の差があらかじめ規定した閾値を超える場合に一方を属性名，もう一方を属性値として扱っている．

林ら[7]は製品性能表に対象を絞り，その場合属性名に用いられるキーワードが限定されることを利用して，まずはキーワードをサンプルデータから学習し，そのキーワードを手がかりにHTML文書内の属性名と属性値を抽出している．

2.4.3. 情報の階層構造

属性名と属性値の組は，半構造化データの中で表現された階層的な情報構造の中に配置されている．図2に階層構造の一例を示す．この階層構造を抽出することも,属性名･値の抽出と合わせて重要な問題である．

このような情報構造は半構造化データの中に間違いなく存在するが，正確に抽出することはきわめて難しい．しかし半構造化データからの情報抽出においては重要な問題であるため,多くの研究が行われている．

先述のWangら[5]は，半構造化データの中に見出すことができた属性名・値の構造の中で，あるサポート値を超えるものを全て抽出しその中から含有関係にあるものを除いた結果を半構造化データ内のSchemaとして求めている．

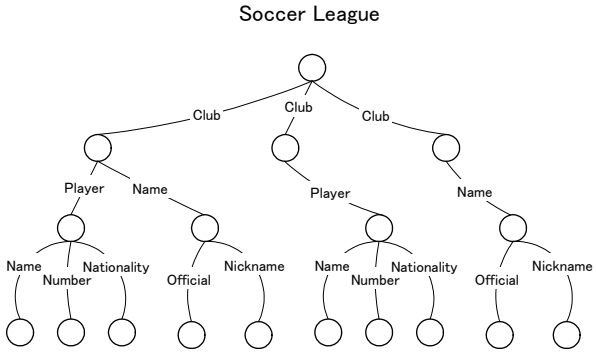


図2　情報の階層構造

2.5. 研究の位置づけ

本研究では，あらかじめURLで特定されたHTML文書内の情報抽出することを目指し，その中でも属性名と属性値の抽出する問題を扱う．

そのために次の仮説を立て，検証する．

- 同一のWebサイトに含まれるWebページの中で，HTMLのタグ構造が似ている文書同士においてはその意味構造も似ている
- 似ている意味構造を持つ文書同士を比較して共通部分と非共通部分を求めれば，属性名と属性値の区別をするためのヒントが得られる

前者についてはいくつか例外となる場合を考えることができる．

例えば同じWebサイトに含まれるHTML文書でも，製作された時期や製作担当者が異なる場合では，HTMLのタグ構造と意味構造を関連付ける内部的なポリシーが変わってしまうこともある．

また一方で異なるWebサイトに含まれるHTML文書の中にも，一部には意味構造が似ているものが存在すると考えられる．

どちらも単純に無視してしまえば情報抽出の有効性においてはマイナス要因となりうるが，全体で考えれば遭遇する頻度が低いため現時点では考慮しない．

3. 頻出パターンと参考文書を用いたHTML文書からの情報抽出

HTML文書内には，さまざまな情報が埋め込まれている．その中でもここでは属性名，属性値のペアに着目し，それらを複数のWebサイトから抽出することを考える．例えば旅行代理店のWebサイトにおけるロンドン行き航空券に関するWebページのURLが指定された場合，対応するHTML文書の中から次のような情

報が抽出されることが期待される.

- **行き先**　ロンドン
- **出発日**　2 月 1 日～4 月 1 日
- **値　段**　100,000 円

この問題に対して，次の手法を提案する.

まず指定された URL を起点に同じ Web サイトに含まれる他の Web ページを取得し，それらの中で HTML のタグ構造が類似しているものを参考文書として保持する.

次に指定された URL に対応する文書(対象文書)と参考文書を比較し，共通部分を「属性名である可能性が高い」部分，非共通部分を「属性値である可能性が高い」部分と考え，それを参考に属性名と属性値のペアを抽出する.

各段階における処理の詳細について以下に述べる.

3.1. HTML 文書のモデル化

半構造化データである HTML 文書を次のようにモデル化する.

- **HTML 文書は，順序木である.** 順序木とは各ノードの子ノードについて，子ノードが複数あった場合に子ノード同士の順序がつけられた木のことである.
- **HTML 文書はラベル付木である.** 各ノードにはラベル(ノードの名前)がつけられている. ラベルは特にノードに固有であるとは限らない.

ラベル付順序木と HTML 文書の対応例を図 3に示す.

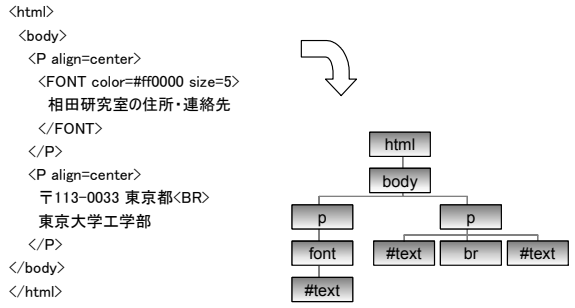


図3　ラベル付順序木による HTML 構造のモデル

3.2. 参考文書の選出

ここでは HTML 文書間の類似度を HTML のタグ構造から求める処理について説明する.

類似度の導出に際しては，HTML タグ名の分布に対して tf/idf 法を用いる. tf/idf 法は文書内に出現するタグの重要度を数値化する手法の一つであり，キーワード検索システムにおいて利用者が入力したキーワードと検索結果として提示されるべき文書間の類似度比較などによく用いられている.

ある文書内でのあるタグの重要度を示す tf/idf 値は，その文書内での重要度と全文書内での重要度を複合して求められる.

あるタグ t_i の文書 D_j における重要度は，出現頻度 *freq(i,j)*を用いて式(1)のように求めることができる. また同様に全文書における重要度は，タグ t_i が出現する文書数 *Dfreq(i)*を用いて式(2)のように求めることができる. D_j に数多く現れるタグ t_i があった場合，その他の文書にはあまり出現しないほうが D_j における重要度は高いと考えられる. そこで前述の式(1)と式(2)を掛け合わせたものをタグの重みとした. 式(3)が最終的に用いた，文書 D_j におけるタグ t_i の重みである.

$$tf_{ij} = \frac{log(freq(i,j)+1)}{log(Terms\ in\ the\ Document\ j)} \quad (1)$$

$$idf_j = log\frac{N}{Dfreq(i)} \quad (2)$$

$$w_j^i = tf_j^i \times idf_i \quad (3)$$

各タグ t_i に関して前述のように定義した重要度を計算していくと，文書 D_x は各タグを次元に割り当てた多次元空間におけるベクトルとして表現することができる. それを式で書くと式(4)のようになる. ここで m は全文書に含まれる全ての異なるタグの数である.

式(4)のように文書 D_x を定義すると，二つの文書間類似度は式(5)のように定義することができる.

式(5)は，ベクトル空間における D_x と D_y のなす角度 θ が小さいほど，類似度が高くなるという考えにもとづいている.

$$D_x = (w_x^1, w_x^2, \cdots, w_x^m) \quad (4)$$

$$sim(D_x, D_y) = cos\theta = \frac{\sum_{k=1}^{N} w_x^k w_y^k}{\sqrt{\sum_{k=1}^{N}(w_x^k)^2} \times \sqrt{\sum_{k=1}^{N}(w_y^k)^2}} \quad (5)$$

このようにしてタグ重要度と類似度を求めることができる. 図 4は，ある HTML 文書についてその Web ページが含まれる Web サイト内の 20 個の HTML 文書を取得し，各文書におけるタグの重要度と文書間距離を求めた結果を示したグラフである. 上のグラフが各文書における各タグの重要度を示したもので，下のグラフが文書 D_1 と各文書 D_1～D_{21} の類似度を示したものである.

このようにして文書間の類似度を計算した上で，その類似度がある閾値を超える文書を参考文書として選出する.

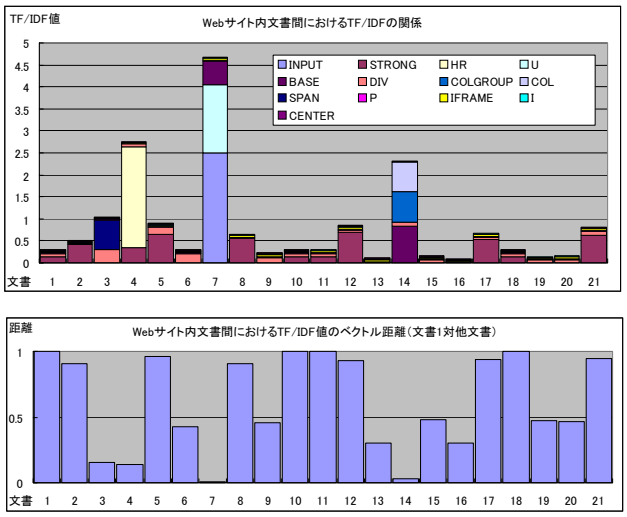


図4　参考文書選出

3.3. HTML 文書内の属性名･値ペア抽出

3.3.1. 文書間の比較

一般的な文書においてその比較を行う問題は，二つの文書 *A*，*B* の最長共通部分(*LCS Longest Common Subsequence*)を求める問題，もしくは最小エディット距離(*SED Shortest Edit Distance*)を求める問題と等価であり，エディットグラフを用いて *SED* を計算する方法が一般的である．

エディットグラフとはそれぞれ要素数が *M*，*N* である文書 A, B の各要素を *x* 軸と *y* 軸上にそれぞれ並べ，それらの交点を縦横の辺で結合し要素同士が等しい場合にのみ*(x-1,y-1)*と*(x,y)*を結合したものである．

そのようなエディットグラフを考えた場合，*LCS* もしくは *SED* を求める問題はエディットグラフ上において次の条件のもとでの*(0,0)*から*(M,N)*までの最小コストの経路を求める問題に還元される．

- 結合された交点間のみ移動が可能
- *(x-1,y)*→*(x,y)*の移動コストは *1*
- *(x,y-1)*→*(x,y)*の移動コストは *1*
- *(x-1,y-1)*→*(x,y)*の移動コストは *0*

HTML 文書はテキストデータであるためそのまま *LCS* や *SED* を求めることもできるが，その場合タグとタグ内テキストを対等に扱うこととなってしまう．そこで次の要領で HTML 文書間の比較を行う．

要素同士の等価条件

- テキストノードでない場合，ラベルの値が完全に一致すること
- テキストノードである場合，テキストの値が完全に一致すること．

HTML 文書の展開条件

- 順序木は，文書の先頭に近いものから順番に深さ優先で列挙したものを比較用シーケンスとする．

図 5に二つの HTML 文書をエディットグラフ上に配置して *LCS* を求める例を示す．

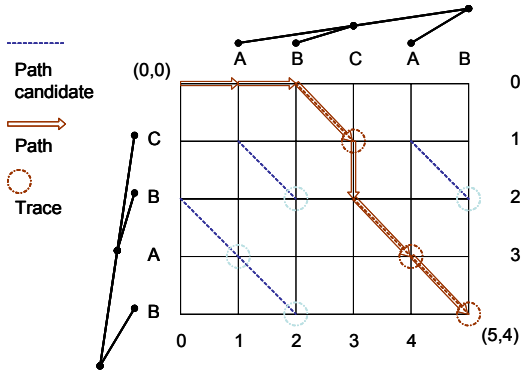


図5　エディットグラフにおける木同士の LCS

この問題は単純に解こうとすると *O(M×N)*の計算量となってしまうが，よく知られている計算方法として *O(ND)*アルゴリズム[9]がある．*O(ND)*アルゴリズムの計算量は *O((M+N)×D)*となる．ここで *D* は最小エディット距離である．この方法で文書同士の比較を行うことで，二文書間の *LCS* が求められる．

3.3.2. 頻出パターンの解析

対象文書内において，HTML タグ木構造の部分木お順に走査し，頻出する部分木を抽出する．そのようにして得られた部分木について，頂点であるノードが兄弟関係にある部分木のグループを抽出する．

最初に抽出された頻出部分木を一つ目の次元，兄弟同士のグループを二つ目の次元と考える，HTML 文書から図 6の要領で複数の二次元マトリックス状のデータが得られる．

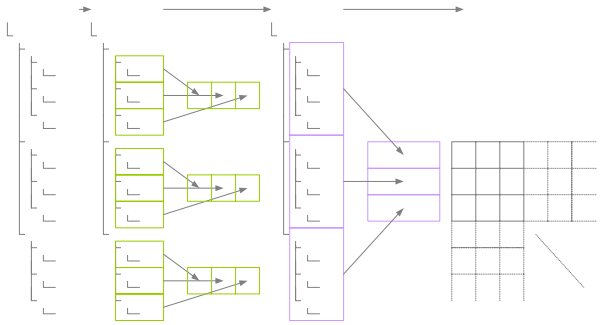


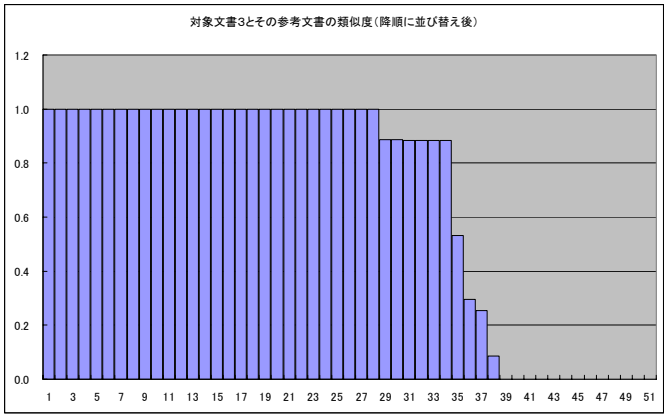
図6　二次元マトリックスとしての情報抽出

二次元マトリックス内の各セルに配置されたデータについて，次の特徴量にもとづいて特徴ベクトルを算出する．

文字属性

次の文字グループに所属する文字列を含むか含まないか, それぞれを 0 と 1 に対応させて特徴量とする．

Lower/Upper/ASCII/Alpha/Digit/Alnum/Punct/Graph/Print/Blank/Space/inBasicLatin/inCJKUnifiedIdeographs/inKatakana/inHiragana



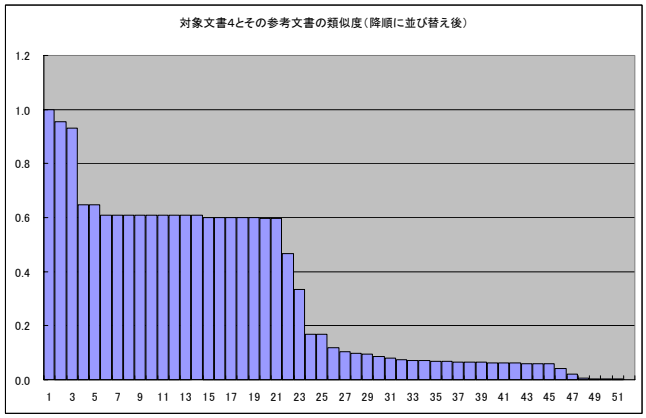


図7　参考文書選出における特徴ベクトル類似度の分布

参考文書との関連性

対象文書と各参考文書間の比較結果にもとづき，*LCS* に含まれるものとそうでないものをそれぞれ 0 と 1 に対応させて特徴量とする．

マトリックス内でのセル位置

属性名は第一行もしくは第一列に配置されることが多いためこの二つの状態であるかないかをそれぞれ 0 と 1 に対応させて特徴量とする．

このようにして求めたセルごとの特徴ベクトルを用いてマトリックス内の行方向及び列方向の類似度変化から，属性名と属性値の判別を行う．

4. 実験

提案した仮説の妥当性，ならびに手法の有効性を検証するために二つの評価実験を行った．それぞれの詳細について説明する．

4.1. 参考文書の選出に関する実験

対象文書が与えられた際に，tf/idf 法とベクトル空間法を組み合わせた提案した手法で参考文書を正確に選出することができるかどうか，次の要領で評価した．

まず異なるジャンルに属する五つの Web サイトを選出し，その中で参考文書の存在が確認できた HTML 文書を対象文書とした．

その対象文書から Web ページ間のリンクを最大深さ 5，最大取得数 50 という条件で探索，参考文書候補を取得した．この候補を対象に提案した手法にもとづいて類似度を評価し，その閾値を 0.8 と 0.9 にした場合についてそれぞれ参考文書抽出の精度と再現率を評価した．

表 6にその結果を示す．全体的に高い精度と再現率で抽出することができたが，対象文書 4 については適合文書を候補選出の時点で一つも取得できていなかった．これは Web サイトの規模と構造が原因で今回設定した探索条件では適合文書まで到達することができなかったためであった．

対象文書 1 における対象文書候補との類似度分布と，対象文書 4 における同様の分布を図 7に示す．適合文書が候補の中に存在しない場合，ここでは正しく選別が行われていることを確認することができる．

表6　参考文書の選出に関する実験結果

	類似度閾値=0.8		類似度閾値=0.9		適合文書数
	再現率	精度	再現率	精度	
対象文書1	0.94	0.94	0.75	0.92	16.00
対象文書2	1.00	1.00	1.00	1.00	18.00
対象文書3	1.00	0.97	0.84	1.00	32.00
対象文書4		0.00		0.00	0.00
対象文書5	1.00	0.96	1.00	1.00	22.00

この結果から，類似度にただ閾値を設けるだけではきれいに適合文書を選り分けることは難しいということが分かる．また参考文書選出の意図を考えると再現率よりは精度が評価指標としては重要であるが，その観点では再現率の低下は無視して類似度閾値を高く設定するべきである．

4.2. HTML 文書内の属性名・値ペア抽出に関する実験

対象文書と対応する参考文書が与えられた上で，提案した手法で属性名と属性値を正確に抽出することができるかどうか，次の要領で評価した．

対象文書とその参考文書は人手で選出した上で提案した手法を適用したが，その際には特徴ベクトルの中に文字属性を入れた場合と入れなかった場合のそれぞれについて，前者をパターン 1，後者をパターン 2 として属性名と属性値ペアの抽出精度，及び再現率を評価した．

表 7にその結果を示す．精度と再現率はともに明らかに低く，まだ手法に改善する余地が大きく残されていることが一目瞭然である．

表7　属性名・値ペア抽出の実験結果

	パターン1		パターン2		適合数	候補数
	再現率	精度	再現率	精度		
対象文書1					16.00	1302.00
対象文書2	0.50	0.82			28.00	66.00
対象文書3					29.00	177.00
対象文書4	0.83	0.25			6.00	56.00
対象文書5			0.74	0.93	10.00	129.00

特徴ベクトルに組み込む特徴量の選出，特徴量相互がもつ影響力の調整などはまだ最適化できているとは言い難い．また今回実際に用いたWebサイトはWWW全体から見ればごく一部に過ぎない．

特徴量の扱いに関して最適なものとなるように工夫を重なる一方で，より多くのWebサイトについて本手法の有効性を試していく必要がある．

5. むすび

年々増加しているWWW上のHTML文書について，単体ではなくて参考文書と合わせて解析することで効果的に情報抽出を行える可能性を指摘し，その観点からtf/idf法，ベクトル空間法，木構造の比較を用いてHTML文書から属性名と属性値のペアを抽出する手法を提案した．

対象文書を起点にWebページ間のリンク構造をたどって取得した参考文書候補からHTMLタグのtf/idf重み付けベクトルを用いて参考文書を抽出する手法は有効であることが示された．ただし現時点では実際に本手法を適用したサンプル数が非常に少ないため，今後より多くのWebサイトを対象に実験を重ねていく必要がある．

一方でHTML文書からの属性名と値のペアを抽出する手法に関しては，まだまだクリアすべき課題が多く残っていることが示された．

はじめに提起した仮説は直感的には妥当だが実験の結果としては裏付けることができなかった．しかしHTML文書内に記述された意味構造を機械的に解釈する上で，他のHTML文書を参考にするという発想は手続きを改良することで有効な手段となりうると考えられる．

今後は参考文書の選出方法，ならびに参考文書を有効活用した対象文書からの情報抽出方法に関して，さらなる改良を加えていきたい．それと並行して，WWW上に存在するさまざまなWebサイトに対して本手法を適用し，その有効性のばらつきなどを検証していきたい．

文　献

[1] R. Kosala and H. Blockeel, "Web Mining Research: A Survey," ACM SIGKDD, July 2000.

[2] S. Abiteboul, "Querying semi-structured data," Proc. ICDT, LNCS, vol.1186, pp.1-18, Springer-Verlag, 1997.

[3] Yoshida, M. "Extracting Attributes and Values from Web Pages," ACL-02 Student Research Workshop, pp. 72-77, 2002.

[4] 丸山 紘平, 上原 邦昭, "半構造データからのマイニングとHMMを用いた情報抽出による知識の相互利用," 電子情報通信学会論文誌 Vol.J85-D1 No.11 pp.1057-1066.

[5] K. Wang and H. Liu, "Schema discovery for semistructured data," In Proceedings of the Third International Conference on Knowledge Discovery and Data Mining (KDD'97), pp.271-274, 1997.

[6] 塚本 修一, 増田 英孝, 中川 裕志, "HTML表データの構造認識システムとその評価," 言語処理学会第9回年次大会, 発表論文集, pp.81-84, 2003.

[7] 林 晃司, 嶋田 和孝, 遠藤 勉, "WWWからの製品性能抽出," 言語処理学会第9回年次大会, 発表論文集, pp.377-380, 2003.

[8] Sudarshan S. Chawathe, Anand Rajaraman, Hector Garcia-Molina, and Jennifer Widom. "Change detection in hierarchically structured information," In Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 493, 04, 1996.

[9] E.W.Myers, "An O(ND) difference algorithm and its variations," Algorithmixa, 1, pp.251-266, 1986.