XML 部分文書検索技術の Web 文書への適用

欅　惇志[†,††]　宮崎　純[†††,†]　波多野賢治[††††]　山本豪志朗[†]　武富　貴史[†]
加藤　博一[†]

† 奈良先端科学技術大学院大学情報科学研究科　〒 630–0192 奈良県生駒市高山町 8916–5
†† 日本学術振興会　〒 102–0083 東京都千代田区麹町 5–3–1
††† 東京工業大学大学院情報理工学研究科　〒 152–8552 東京都目黒区大岡山 2–12–1
†††† 同志社大学文化情報学部　〒 610–0394 京都府京田辺市多々羅都谷 1–3
E-mail: †{atsushi-ke,miyazaki,goshiro,takafumi-t,kato}@is.naist.jp, ††khatano@mail.doshisha.ac.jp

あらまし　本稿では，XML 部分文書検索技術の Web 文書 (HTML 文書) への適用を行う．XML 部分文書検索技術では，検索結果としてユーザの情報要求を満たす箇所そのものを提示し，検索時のユーザの負担を軽減することを目指す．XML 部分文書検索技術を適用する上で，HTML 文書の特徴である，1) 文書内容の論理構造と文書の物理構造が一致しないため，部分文書検索において利用する文書構造を適切に用いられない可能性がある，2) 文書中にサイトの目次やリンク集などのように直接ユーザの情報要求を満たさない箇所を多分に含むことにより生じる問題を解決する必要がある．1) に対して，潜在的に文書の構造情報を表現すると考えられるタグの情報を利用した，論理構造に沿った文書構造への再構造化手法を提案する．また，2) の問題には，目次やリンク集などのハイパーリンクを多分に含むという特徴に着目したサブ・コンテンツフィルタの提案を行う． 評価実験の結果，再構造化によってより適切な粒度の部分文書を抽出できたことによる検索精度の向上と，サブ・コンテンツフィルタによって文書中のメインコンテンツ以外の箇所の除外に成功した．また，部分文書検索を行うことで，文書検索と比較して再現率が低下するものの，より高精度な検索を実現できることが判明した．

キーワード　XML 情報検索，部分文書検索，Web 文書，構造整形，サブコンテンツ，フィルタ

1. はじめに

XML 部分文書検索の検索単位は XML 文書 (構造化文書) 中の一部分，即ち部分文書(注1) (element) である．そして XML 部分文書検索の目的は，文書中からユーザの求める情報を含む部分文書を特定して提示することである．そのため，ユーザは自らが求める情報が文書中のどの部分に記述されているのかを探す必要がなく，XML 部分文書検索システムを利用することで，検索におけるユーザの負担を軽減することが可能である．

XML 部分文書検索に関する研究では，これまで，学術論文や Wikipedia 記事などの XML 文書を対象として取り組まれてきた．その結果，高精度な XML 部分文書検索や，高速な XML 部分文書検索が実現しつつある．そして，文書中の構造情報を利用して適切な検索結果の特定を行う XML 部分文書検索技術は，XML 文書以外の構造化文書への適用も期待される．中でもとりわけ，構造化文書の中でも最も一般的に利用される Web 文書 (HTML 文書) へ適用することで，XML 部分文書検索技術の適用範囲と利便性は格段に飛躍することが見込まれる．

HTML 文書に対して XML 部分文書検索技術を適用する上で，使用用途の違いから，XML 文書と HTML 文書ではその特性や性質が異なる．以下に HTML 文書の特徴を列挙する．

（1） タグの整合性が保たれていないことが多い．

（2） 文書内容の論理構造と文書の物理構造が一致しない可能性がある．

（3） 情報要求を満たさない箇所 (部分文書) を多く含む．

まず，(1) に関して，構造化文書に対して部分文書検索技術を適用する際には，各部分文書を正しく取り扱うことが可能なように，開始タグと終了タグの関係に整合性が保たれている (開始タグと終了タグの対応が取れている) 必要がある．一般的に，データの管理に用いられる XML 文書においては，正しく整合性が保たれていることが多く，整合性が保たれていない XML 文書を読み込めない XML パーサーも多く存在する．その一方で，ブラウザ上での表示を目的として用いられる HTML 文書は，タグの整合性が厳密に保たれていなかった場合でもブラウザが自動で解釈して表示可能な場合が多いため，タグの対応が取れていない HTML 文書が多数存在する．このような問題を解決するため，HTML 文書をパーシングする際には HTML 文書のタグの整形ツール(注2) が用いられることが一般的である．

(2) の特徴に関して，人手によって記述される文書の多くは論理的な構造を持つ．例えば，文書全体は複数の章で構成されており，同様に，各章は複数の節から，各節は複数の小節から，小節は複数の段落から構成される．これらの章立て情報は文書の内容を理解する上で大きな手助けとなる．前述の学術論文や Wikipedia 記事では，これら論理的な構造に沿って文書の物理的な構造もタグで定義されており，部分文書検索ではこれらの

(注1)：部分文書については 2.2 節にて後述する．

(注2)：CyberNeko HTML Parser (http://nekohtml.sourceforge.net/)

文書の構造を利用して，情報要求を満たす内容が記述された粒度の部分文書を特定してユーザへ提示する．つまり，部分文書検索を適用する上では，文書の物理構造が，文書内容の論理構造に基づいて定義されている必要があるということである．

それに対して，HTML 文書を構成する HTML タグの中には，テキストのフォントサイズやフォントカラーなどの変更など，テキストの強調を目的としたタグが大部分である．その結果，HTML 文書の物理構造は文書内容の論理構造が十分に反映されておらず，このような状況では適切に XML 部分文書検索技術を適用することができない可能性がある．

(3) に関して，HTML 文書は複数のパートから構成されることが多い．例えば，Weblog (ブログ) のエントリの文書構造の例を図 1 に示すと，ブログタイトル，本文 (メイン・コンテンツ)，過去のエントリ一覧，リンク集，エントリのカテゴリ分類，及び，タグクラウドなどの複数のパートから構成される．このとき，メイン・コンテンツ以外のパート (以降，サブ・コンテンツと呼称する) は，他のエントリ (ページ) へのリンクとして存在するために，探索的情報検索においては有効であると考えられるものの，これらの部分に記載された情報のみでユーザの情報要求を達成することは考えにくい．そのため，これらのパートが検索結果の上位に配置されることは適切ではないが，索引語の重み付けベースのスコアリング手法においては，サブ・コンテンツ中でクエリキーワードが頻出した場合に高いスコアが付与される傾向にある．従って，ユーザの情報要求を満たす部分文書のみを検索対象として扱うために，文書中からメイン・コンテンツ以外のパートを特定し，検索結果から除外することが必要である．なお，サブ・コンテンツとしては上記以外にも，著者のプロフィールやカレンダー，検索ボックスなども該当するが，本稿においては特に，有用な検索結果の提示に悪影響を及ぼすパートのことをサブ・コンテンツ指すとする．

前述の特徴に起因する課題を解決すべく，本稿では，部分文書検索のための文書の整形・加工を行う．その際，1) 文書の論理構造に沿った構造への変換を目的として，文書の再構造化と，2) サブ・コンテンツによって構成される部分文書の除去を行う．

1) に関して，過去の研究において，HTML タグの中にはトピックの分断を表すタグが存在する [9] ということが報告されているために，本稿ではそれらの研究で得られた知見をもとに，論理構造を反映させた文書構造への再構造化を行う．

2) の解決においては，過去のエントリ一覧やリンク集など，他のページ中へのリンクを目的として記述される箇所にはアンカータグ (A) が頻出することが予測されることを利用して，サブ・コンテンツから構成される部分文書を除外するためのフィルタを提案する．

2. XML 部分文書検索に関する基本事項

本節では，文書検索及び関連技術に対する XML 部分文書検索の位置づけと，部分文書検索の概要を説明する．

2.1 XML 部分文書検索技術の位置付け

XML 部分文書検索における最大の関心は，クエリに適合する箇所，即ち部分文書を抽出し，それらをクエリに対する適合度の高さの降順でユーザへ提示することである．多くの Web

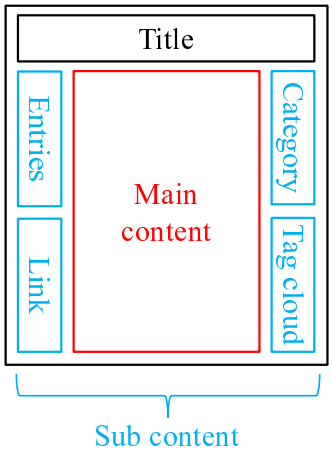

図 1　複数のパートから構成される Weblog 記事

検索システムがクエリに適合する文書のリストを提示するのに対して，XML 検索システムはクエリに適合する部分そのもののリストを提示することができる．これにより，ユーザは文書中から情報要求を満たす部分を自ら発見する必要がなくなるため，情報検索を行う際のユーザの負担を大きく軽減することが可能である．

関連する技術として，スニペット [12] やパッセージ検索 [4] が存在する．多くの Web 検索システムは，検索結果としてクエリに適合する文書のリストを提示する際に，スニペットと呼ばれる 150 文字前後の要約文も併せて提示する場合が多い．スニペットはクエリキーワードとその周辺のテキストを抽出する技術であり，検索結果中からいずれの文書が閲覧するのに適切であるかをシステム利用者が判断するための要約文である．インターネット利用者に対して行った Web 検索の信頼性に関する調査 [13] では，多くの検索システム利用者がスニペットを利用していることが報告されている反面，必ずしも満足な結果が得られているわけではないということも報告されている．これは，スニペットは文章の文脈を考慮せずに生成されるために，スニペット単体で理解不能な場合も多いためである．このことから，検索システム利用者はスニペットのみから情報要求を満たすことはできず，結局のところ文書検索システム利用者は文書を閲覧し，必要な情報を自ら発見しなければならない．

また，パッセージ検索 [4] も XML 部分文書検索と同様に，文書中からクエリに適合する箇所のみを抽出することを目的とした情報検索技術である．その際の情報単位として，XML 部分文書検索と同様に HTML 文書中のタグを利用する場合や，固定のウィンドウ幅を設定する場合や，同一のトピックで記述された箇所を特定するアプローチなどが存在する．XML 部分文書検索では文書構造を利用した索引語の重み付けを行っている点や，文書中の粒度の観点を考慮しているという点で，パッセージ検索とは異なる．

2.2 部分文書とその重複関係

部分文書の概要について説明するために，図 2〜図 4 を用いて具体例を示す．

まず，図 2 は文書番号 (DocumentID, DocID) 1 の XML 文書の例であり，図 3 は図 2 の XML 文書を木構造で表現した

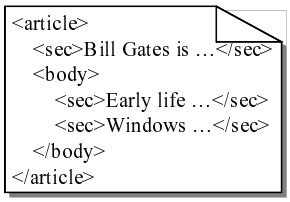

図 2　XML 文書

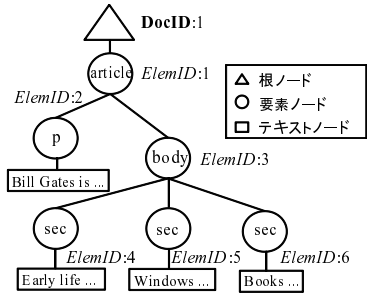

図 3　XML　木

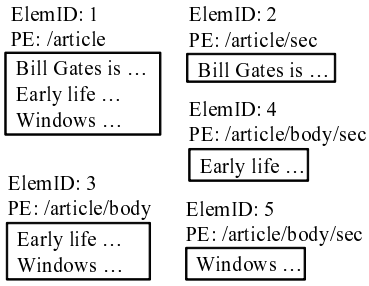

図 4　部 分 文 書

図である．構造化文書は一般的に木構造で表現することができ，文書構造の視認性の向上を目的として度々木構造で表現される．本稿においても同様に，XML 文書を木構造と見立てて議論を進めることとする．このとき，XML 文書のそれぞれの開始タグと終了タグが XML 木の各要素ノードのノード名に対応しており，タグの入れ子は要素ノードの親子関係によって表現されている．また，各要素ノードに対して，文書順に部分文書番号 (ElementID, ElemID) を割り振る．図 4 の各部分文書は，図 3 の XML 木の各要素ノード以下に含まれるテキストノードを結合した文字列と対応する．つまり，文書全体を表す `article` ノードと対応する部分文書は子孫に存在するテキストノード全てを結合した文字列であり，`body` ノードと対応する部分文書はその子ノードである三つの `section` ノードに含まれるそれぞれのテキストノードを結合した文字列である．包含関係 (先祖 · 子孫関係) を持つ部分文書間においてテキストノードの重複が発生するのはこのためである．なお，ある要素ノード (部分文書) に着目した際に，先祖にあたる要素ノードを粒度の大きなノード，逆に子孫にあたる要素ノードを粒度の小さなノードと表現する．

仮にユーザが “Early life …”，“Windows…” に関する情報を求める場合，XML 部分文書検索では ElemID 3 の部分文書を提示する．これは，部分文書中にユーザが求める情報全てを含み，余分な情報を含まないためである．このように，ユーザの情報要求に対し必要十分な情報を提示することが XML 部分文書検索における高精度検索である．

2.3　部分文書検索における検索結果の提示方法

同一文書中の部分文書間には包含関係があるため，部分文書検索の枠組みでは検索結果として抽出される部分文書間で重複が発生する可能性がある．いくつかの先行研究では重複が発生することを考慮せずに，クエリ処理によって算出される部分文書の適合度に従ってランキングされた順位付きリストを提

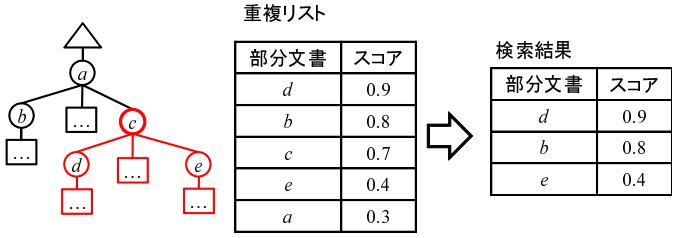

図 5　部分文書抽出

示していた．本論文では，このようなクエリ処理によって得られた，重複を排除するために特別な処理を行っていないリストを**重複リスト** と呼ぶこととする．そして，文献 [6] において重複リストを用いた場合の検索精度の低下が報告されて以来，多くの研究ではリスト中から重複を排除した**非重複リスト**を用いている．現存する最大の XML 検索のためのプロジェクトである INEX (INitiative for Evaluation of XML retrieval) project(注3) の XML 情報検索用のトラック全般 [2] においても非重複リストを使用している(注4)．

重複リストから非重複リストを作成する際にはいくつかのアプローチが採用されており，主なアプローチを以下に列挙する．

（1） 部分文書のスコアの降順に，重複が起こらないように一つの文書から一つもしくは複数の部分文書を抽出する

（2） 部分文書の統合による検索結果再構成する [17]

（3） 文書全体を提示する

(1) に関して，単に適合度の降順に，重複が起こらないように部分文書を抽出した場合には，適合箇所を取りこぼす可能性が存在する．図 5 を用いてその具体例を示すと，このとき，赤く着色された，ノード c を根とする完全部分木と対応する部分文書 (以降，ノード x を根とする完全部分木と対応する部分文書のことを単に x と表現する) が適合部分であるとする．重複リスト中の部分文書を，スコアの降順に重複が生じないように抽出を行うと，まずは d が抽出される．続いて b が抽出され，その後に c の抽出が試みられるが，c は既に抽出されている d と重複が発生するために，c は抽出されずに破棄される．その結果，全ての適合部分を網羅して抽出できない．

そこで，我々は (2) の部分文書の統合による検索結果再構築手法の提案を行った [17]．各部分文書に付与された得点やテキストサイズ，更に部分文書間に存在する包含関係に着目して部分文書の統合を行い，検索結果の再構成を行う．その概要については次節にて述べる．

なお，(3) のアプローチでは適合箇所を網羅的に抽出することが可能であるものの，文書中のクエリと適合しない箇所も取り出す可能性がある．また，文書全体を検索結果として提示することとは，即ち，文書検索と同義であるため，部分文書検索の本来の趣旨からも外れ，適切なアプローチとは考えにくい．

3.　関 連 研 究

本節では，XML 部分文書検索において用いられる，高精度

(注3)：`http://www.inex.otago.ac.nz/`

(注4)：各クエリに対して上限 1,500 件まで，抽出すべき部分文書が存在する限り抽出し提示する．

検索実現のための諸技術について説明する．また，HTML 文書の内容理解に関する関連研究について述べる．

3.1 高精度 XML 検索に関する関連研究

XML 部分文書検索において適合部分文書を発見する際は，まず各部分文書中の索引語に対して任意のスコアリング手法によって重みを算出し，そしてそれらの重みからクエリと適合する部分文書を特定するアプローチが主流である．

XML 部分文書検索において利用されるスコアリング手法の多くは，文書検索用のスコアリング手法を基にしている．文書検索と部分文書検索のスコアリング手法における最も大きな違いは，大域的重みと分類される統計量の算出方法である．文書検索においては全ての文書は同じ属性を持つと見なして，全ての文書を母集団として大域的重みを算出する．それに対して部分文書においては，同じ属性を持つと見なす部分文書群に分類し，それぞれを母集団として大域的重みの算出を行う．属性の分類は，全ての部分文書は同じ属性を持つとするアプローチ，同じタグで定義される部分文書は同じ属性を持つとするアプローチ，そして同じ path 式を持つ部分文書は同じ属性を持つとするアプローチなどが存在する．

部分文書検索システム評価のための大規模なテストコレクションである INEX Wikipedia テストコレクション [3] においては，同じ path 式を持つ部分文書は同じ属性に分類して統計量を算出するアプローチが最も高精度と報告されている [15] ものの，統計量算出に用いる文書数が十分に確保されない状況においては path 式ベースの大域的重み算出では適切に統計量を算出できないということが報告されている [18].

部分文書検索用のスコリング手法としては，TF-IPF [10], BM25E [11]，部分文書検索用クエリ尤度モデル [14] など多数存在し，過去の研究において，BM25E が最も高精度検索に適切であるという結果が得られている [8].

また，我々は文書集合全体からいかなるクエリに対しても解にならない部分文書を選定する研究 [16] や，検索結果としてより適切な粒度を決定する研究 [7] に取り組んできた．これらの研究で得られた知見として，文書全体などの大きな粒度の部分文書にはユーザの情報要求に適合しない箇所を含む可能性があり，逆に，小さな粒度の部分文書はそれ単体ではユーザの情報要求を満足させることができず，部分文書の粒度としては中程度の粒度が適切であるということが判明した．なお，小さな粒度の部分文書を特定する際には，部分文書中の索引語数による閾値の設定が容易かつ効果的である．

更に，我々は，文書中から検索結果として最適な粒度の部分文書を特定するため，重複リストから非重複リストを作成する際の部分文書統合による検索結果の再構成手法を提案した [17]. ここで，最適な粒度の部分文書とは，文書中のクエリに対する適合箇所を最大限含み，非適合な記述を極力含まない部分文書のことであるとする．その際のアプローチとして，既存研究において重複リストから非重複リストを作成する際には各部分文書のスコアの優劣で抽出する部分文書を決定していたが，再構成手法では，部分文書の粒度の大小によって抽出する部分文書を決定する．つまり，包含関係を持つ部分文書のうち，より多くの適合箇所を含む，大きな粒度の部分文書を検索結果として抽出する．ただし，大きすぎる粒度の部分文書を抽出した場合にはクエリに対して非適合箇所を含む可能性があるため，一つの文書から抽出可能なテキストサイズの上限に制限を設ける．

3.2 HTML 文書の内容理解に関する関連研究

構造化文書中のタグは大別して 1) 独立したデータを囲むタグと，2) 文を途中で切ることがあるタグ，と定義される [19]. 本論文では，1) のタグとは構造を表すタグであり，2) のタグとは修飾や属性，特定のコンテンツを表すタグであると見なす.

1) の構造を表すタグに関して，HTML タグのうち，従来から HTML 文書中で使用されてきた `HTML` タグや `HEAD` タグ，`TITLE` タグ，`BODY` タグ，及び，`P` タグに加えて，HTML5[(注5)] から新たに追加されたタグのうち，一つの記事に対して付与する `ARTICLE` タグ，一纏まりの内容を意図する `SECTION` タグ，目次などナビゲーション用要素のための `NAV`，そしてサブ・コンテンツを記述するための `ASIDE` タグなども該当する．また，表 (`TABLE`) やリスト (`UL`, `OL`, `DL`) など特定のコンテンツを表すタグも構造を表すタグに分類されるとする．

なお，入れ子による定義を行うことが可能な `SECTION` タグをはじめとして，新規追加されたタグを適切に利用することで，文書の論理構造を適切に表現した物理的な文書構造を定義することが可能となったものの，未だ HTML5 に準拠した HTML 文書の普及率は決して高いとはいえず，本稿においてもこれら新規追加されたタグは利用できない状況においても適用可能なアプローチを提案する．

また，2) のタグにはテキストのスタイルや色や大きさを指定する `B` タグ，`FONT` タグ，`I` タグなどが該当する．HTML タグにおいては 2) のタグが大部分であり，部分文書検索において適切な粒度の部分文書を特定する上で重要な手がかりとなる構造を表すタグは少数である．

続いて，HTML タグの中にはトピックの分断を意図するタグが存在すると報告されている [9]．具体的には Heading タグ (`H1`–`H6` タグ) や `HR`, `BR` タグなどであり，文献 [9] において，これらのタグをトピックの分割点として見なし，コンテンツの分割に利用している．

その他の HTML 文書中のコンテンツの分割手法として，HTML 文書のブラウザで表示した際の見た目に基づく分類手法が存在する [1]．手法の利点として，本来の文書構造では同一の部分に分類されないコンテンツでも，ブラウザ上での表示における類似度が十分に高ければ，同一のコンテンツとして分類することが可能である．

4. 部分文書検索技術の適用のための取り組み

HTML 文書の特徴によって生じる，XML 部分文書検索技術適用の際の問題点を解決すべく，1) 論理構造に基づく文書の物理構造への再構造化手法と，2) サブ・コンテンツを除外するためのフィルタの提案を行う．

4.1 文書の再構造化

文書内容の論理的な構造は，章，節，小節，段落など複数の

(注5)：`http://www.w3.org/TR/html5/`

粒度の章立てから構成されるものの，HTML 文書において利用可能な構造情報は，実質のところ，入れ子による定義を行うことができない P タグによって表される段落の粒度のみである．その結果，HTML 文書においては，文書内容の論理構造と物理的な文書構造が一致していないという状況が発生する．

その一方で，HTML タグの中にはトピックの分断を意図すると報告されているタグが存在する [9]．H1–H6 の Heading タグは，内容の重要度に併せて Heading のレベル (タグ中の数値) が設定される(注6)ために，論理的な文書構造を表現していると考えられる．これは，重要度の高い見出しほど大きな粒度のトピックを表し，重要度の低い見出しほど小さな粒度のトピックを表すと見なすことができるためである．

Heading タグの開始タグと終了タグ中に含まれる情報は見出しのタイトルであるのに対して，Heading タグの直後のテキストの内容は見出しのタイトルに関する内容が記述されていることが一般的であるため，Heading タグの情報によって，文書に論理構造に基づいた物理構造を付与可能であることが期待される．従って，我々は Heading タグとそのレベルをもとに，Heading Container タグ (HC1–HC6) を挿入することで文書の再構造化を行う．Heading Container タグの開始タグと終了タグの挿入ルールはそれぞれ以下の通りである．

開始タグ: Heading タグ (Hx) が出現すれば，Heading タグの直前に Heading Container タグ (HCx) を挿入する．

終了タグ: 新たに出現した Heading タグ (Hx) のレベルが，それ以前に出現した Heading タグ (Hy) と同等もしくは大きなレベルの場合，新たに出現した Heading タグの開始 Heading Container タグ (HCx) を挿入する直前に Heading Container タグの終了タグ (HCy) を挿入する．ただし，BODY タグはいずれの Heading タグよりも大きなレベルとして扱う．つまり，BODY タグの終了タグの直前にまだ閉じられていない全ての Heading Container タグの終了タグを挿入する．

具体例を用いて処理を説明する．図 6 の左の元の HTML 文書中において，Heading タグは H2，H3，H3，H1 の四つが出現する．まず H2 タグが出現した際には，H2 タグの直前に HC2 の開始タグを挿入する．次に，H3 タグが出現しており，これは先ほど出現した H2 タグよりも小さなレベルであるため，終了タグは挿入されず，H3 タグの直前に HC3 タグが挿入される．続いて出現する Heading タグは H3 タグであり，直前に出現したタグと同等のレベルであるため，HC3 タグの終了タグが挿入され，HC3 タグの開始タグが挿入される．最後に出現する Heading タグは H1 タグであり，既に挿入されている H2 タグ及び H3 タグのレベルよりも大きい．従って，順次 HC3·HC2 タグの終了タグが挿入された後に HC1 タグが挿入される．また，BODY の終了タグの直前に HC1 の終了タグが挿入され，文書の再構造化処理が完了する．

4.2 サブ・コンテンツを除去するためのフィルタ

情報要求を満たすとは考えにくい，過去のエントリ一覧やリンク集，もしくはエントリのカテゴリ分類，タグクラウドなどのサブ・コンテンツから構成される部分文書の除外を行うためのフィルタの提案を行う．

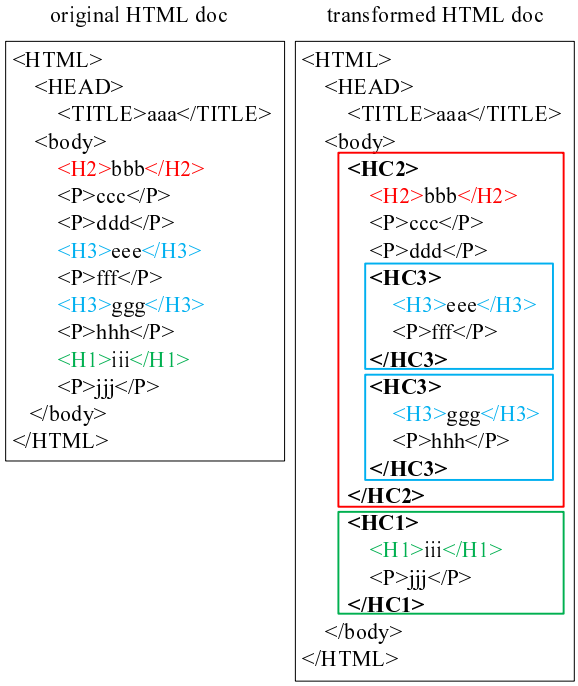

図 6　文書の再構造化処理

学術論文や Wikipedia 記事を対象とした XML 部分文書検索において提案されてきた，不要な部分文書を特定するための研究の知見として，索引語数が小さすぎる部分文書は検索結果として不適切であるということが報告されている [16]．それに対して，ここで除外を目指す部分文書は，必ずしも索引語数が小さいとは限らない．従って，これまでの索引語数に着目したアプローチとは異なる方法によって，前述の不要な部分文書の特定を行う必要がある．

前述のサブ・コンテンツの特性を考えた場合に，これらのページの多くは他のページへのハイパーリンクが存在する．従って，これらの部分文書の中には A タグが多く含まれていることが予想されため，A タグに着目したサブ・コンテンツ抽出方法について議論する．A タグの出現の多さを測る指標として考えられるものを以下に列挙する．

（1） **A タグの出現確率:** 部分文書のテキストサイズによって正規化された A タグの出現頻度

（2） **木構造中の A タグで定義されるノードの比率:** 部分文書の子孫ノードのうちの A タグで定義されるノードの割合

（3） **A 中のテキストサイズの比率:** 部分文書のテキストサイズに対する，A タグに含まれるテキストの割合

まず，(1) に関して，テキストサイズに対する A タグの出現頻度が高いからといって必ずしもサブ・コンテンツとは言い切れない．例えば，メイン・コンテンツ中に複数の画像が存在しており，それらの画像からはオリジナル画像などへのリンクが存在する場合に，テキストサイズに対する A タグの出現頻度が高くなり，サブ・コンテンツであると誤判別される可能性がある．従って，A タグの出現確率は必ずしも常に適切な指標であ

(注6)：重要な項目の見出しから順番に H1 タグから H6 タグを割り振る．なお，本稿では Heading タグの数字が小さいほど大きなレベルのであると表現する．

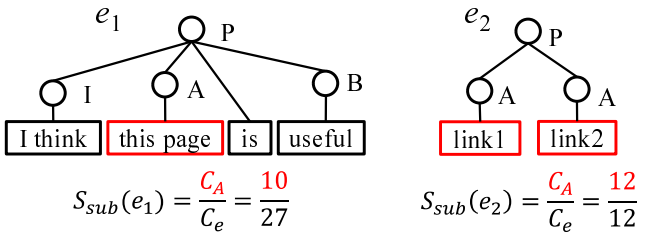

図 7　サブ・コンテンツスコアの算出

るとはいえない．

また，(2) では，部分文書中に出現するタグのうち A タグの割合が大きい場合にサブ・コンテンツであると判定するが，HTML 文書中のテキストが必ずしもマークアップされているとは限らない．当然ながらメイン・コンテンツ中においても他のページへのハイパーリンクは存在し，それらの影響によって誤判別が発生する可能性がある．そのため，(2) も不適切である．

これらに対して，(3) は A タグ中のテキストをカウントするため，別ページへのハイパーリンクとして利用される A タグのみを考慮可能である．また， A タグ以外のタグの生起状況に依存せずにサブ・コンテンツ判別が可能であるため，誤判別なくサブ・コンテンツを発見することが可能であると考えられる．

以上より，部分文書のテキストサイズに対する A タグ中のテキストの比率によるサブ・コンテンツ判別を行う．部分文書 e のサブ・コンテンツスコア $S_{sub}(e)$ は以下の数式で表される．

$$S_{sub}(e) = \frac{C_A}{C_e} \tag{1}$$

ただし，C_A は e の子孫ノードのうち，A タグの直下のテキストノードのテキストサイズの合計，C_e は e のテキストサイズとする．なお，$S_{sub}(e)$ が閾値 $\tau (0 \leqq \tau \leqq 1)$ を超える部分文書はサブ・コンテンツとして検索結果から除外される．

サブ・コンテンツスコアの算出例を図 7 に示す．このとき，閾値 τ が 0.5 であったとする．図中左の e_1 は部分文書長が 27 文字の部分文書であり，A タグで定義されるノード以下のテキストサイズは 10 文字である．従って，$S_{sub}(e_1) = \frac{10}{27} = 0.37$ となり，e_1 はメイン・コンテンツであると判定される．それに対して e_2 は部分文書中のテキストは全て A タグの直下に存在するため，$S_{sub}(e_2) = \frac{12}{12} = 1.0$ となる．その結果，e_2 はサブ・コンテンツであると判定される．

5. 評価実験

本節では，まず，提案した再構造化手法の影響の確認とサブ・コンテンツフィルタの効果の確認と閾値の設定を行う．更に，それらを踏まえて，HTML 文書への部分文書検索技術適用の検索性能を検証する．

5.1 実験環境

以降の評価実験では，Web 文書 (HTML 文書) と Web 検索システムに対して利用されるクエリを用いて実験を行う．その際，国立情報学研究所主体で開催される情報検索のための国際プロジェクトである NTCIR-10 のタスクの一つである One Click Access task [5] (1CLICK-2) タスクが提供するテストコレクションの文書集合とクエリ集合を利用した．

表 1　追加された Heading Container タグ

タグの種類	タグの個数 (%)
HC1	33217 (.092)
HC2	113300 (.31)
HC3	116787 (.32)
HC4	60276 (.17)
HC5	26819 (.074)
HC6	10880 (.030)

表 2　構造の深化度

深化度	文書度 (%)
0 (変化なし)	3639 (.14)
1	5194 (.19)
2	8300 (.31)
3	6912 (.26)
4	2509 (.093)
5	346 (.013)
6	28 (.0010)

1CLICK-2 のクエリは，Web 検索システムに問い合わせられたクエリログから抽出された 100 個のクエリであり，クエリは八つのタイプ (ARTIST, ACTOR, POLITICIAN, ATHLETE, FACILITY, GEO, DEFINITION, QA) に分類される．

また，文書集合は各クエリを Web 検索システムへ発行したときに得られる上位 500 件の文書集合である．

実験で使用した計算機は，CPU: Intel Xeon X7560 (2.3GHz) を 4 つ，512GB のメモリ，4.5TB のディスクアレイ装置 (1TB SATA HDD×12 の RAID 1 構成，計算機とは 4Gbps FC による接続) を搭載する．OS は Oracle Enterprise Linux 5.5 であり，実験システムは GNU C++ を用いて実装を行い，検索用の索引は BerkeleyDB 5.3 を利用して構築した．

5.2 文書の再構造化に関する調査

文書の再構造化によってどの程度文書構造が変化したのか調査を行う．表 1 は追加された Heading Container タグの種類と，その個数及び全体に対する割合を表す．各項目は文書中での Heading タグの出現数と対応しており，Web 文書中には H2 タグと H3 が特に利用されていることが分かる．また，一文書辺り平均 13.4 個の Heading Container タグが追加された．

更に，Heading Container タグが追加されたことでどの程度文書構造が深くなったのか (深化度) を計測した．例えば，文書中で H2 タグと H3 タグが出現すれば深化度 2 であり，H1 タグから H6 まで順番に出現した場合には深化度 6 である．表 2 は，文書の深化度とその文書数を表す．8 割以上の文書中において Heading タグが出現し，3 段階の構造の深化が行われることが最も多く，平均で 2.23 段階深化したという結果が得られた．

以上の結果から，Heading タグに着目することで，元来は段落の粒度の構造のみを持つ HTML 文書に対して新たな粒度を付与することが確認された．ただし，これは同時に検索対象部分文書の増加を意味し，元の文書と比較して検索対象が 25% 増加した．Web スケールなど，大規模な文書集合に対して手法を適用する上では，部分文書の爆発を抑制するアプローチを考案する必要がある．

5.3 サブ・コンテンツフィルタに関する実験

サブ・コンテンツフィルタによって，文書集合中からサブ・コンテンツ，即ち，過去エントリの一覧やリンク集などから構成される部分文書を判別することが可能であるかの確認を行う．

まず，部分文書のサブ・コンテンツスコアごとに，サブ・コンテンツの判別精度を計測する．その際，サブ・コンテンツスコアを 0.1 から 0.2 刻みで分割し，各項目ごとに 10 個の部分文書を無作為抽出し，各部分文書がサブ・コンテンツであるか

表 3　サブ・コンテンツフィルタの判別精度

サブ・コンテンツスコア	判別精度
0.1〜0.3	50%
0.3〜0.5	60%
0.5〜0.7	80%
0.7〜0.9	100%
0.9〜	100%

どうか手作業で評価した．

表 3 の通り，サブ・コンテンツスコアが 0.7 よりも大きい部分文書は全てサブ・コンテンツであった．この結果から，サブ・コンテンツフィルタの閾値 τ には 0.7 を設定する．なお，サブ・コンテンツフィルタの適用によって，全体の 9% の部分文書がサブ・コンテンツとして除外される．

5.4　HTML 文書への XML 部分文書検索技術適用の効果

続いて，検索性能を計測することで HTML 文書に対する部分文書検索技術の有用性を検証する．

5.4.1　実 験 準 備

XML 部分文書検索技術の Web 検索への適用の可否を確認するにあたり，本稿の実験では比較的部分文書検索との相性のよいと考えられるカテゴリのクエリに対して評価を行う．つまり，明確な検索対象や取り出すべき事実が存在するクエリというよりむしろ，クエリに関連する情報を網羅的に抽出することを目指すクエリを多く含む DEFINITION カテゴリのクエリ 15 個を対象として評価を行った．これは，これまで部分文書検索は文書中からクエリに適合する情報を網羅的に抽出することを目標として取り組まれてきており，我々が過去に開発した技術もこの目標に準拠しているためである．

実験の手順として，まず，文書集合中の各部分文書に対して BM25E [11] によってスコアリングを行う．ただし，本稿の実験で利用した文書集合は，path 式ベースによる大域的重み算出においては文書数が十分ではない可能性があるため，タグの種類ベースの大域的重みを採用する．また，過去の高精度検索実現を目標した研究 [7], [17] の知見をもとに，索引語数 15 未満の部分文書を除外し，検索結果中に重複する部分文書が存在すれば部分文書の統合処理を行った．

続いて，評価方法について説明する．検索結果として提示された部分文書を含む文書の中からクエリに対する適合箇所を網羅的に抽出する．その際，情報の重要度の評価は行わず，クエリに対して適合するかどうかの評価を手作業において行った(注7)．

評価尺度としては，検索精度 (precision) と再現率 (recall)，及び F 値 (F–measure) を計測した．一般的な文書検索の評価では各文書は適合もしくは非適合の二値による評価が行われることが(注8) のに対して，部分文書検索では文書中の適合箇所をどの程度抽出できているのかによって評価する．検索精度と再現率，及び F 値は以下の式で算出される．

(注7)：我々のグループは 2002 年から 2010 年まで INEX project に参加し，テストコレクションの評価データの作成にも携わった．これらの経験から，極力 INEX project によって用いられてきた手順と方法に準拠して評価を行った．

(注8)：nDCG のように適合度を多段階に評価する指標も存在する．

表 4　検索精度と再現率，F 値による検索性能評価

	@1			@5			@10		
	精度	再現率	F 値	精度	再現率	F 値	精度	再現率	F 値
部分文書	.503	.233	.176	.498	.305	.298	.512	.351	.344
再構造化	.517	.282	.230	.545	.372	.424	.586	.360	.418
文書検索	.382	1	.458	.372	1	.429	.394	1	.549

表 5　検索結果のテキストサイズの平均値と標準偏差

	平均テキストサイズ	標準偏差
部分文書	2288.754	2576.560
再構造化	1717.870	1292.166
文書検索	6411.987	1486.383

$$precision = \frac{\text{部分文書中の適合箇所のテキストサイズ}}{\text{部分文書のテキストサイズ}} \quad (2)$$

$$recall = \frac{\text{部分文書中の適合箇所のテキストサイズ}}{\text{文書中の適合箇所のテキストサイズ}} \quad (3)$$

$$F - measure = \frac{2 * precision * recall}{precision + recall} \quad (4)$$

なお，多くの情報検索の評価尺度の再現率とは文集合全体のうちの適合箇所の抽出割合，いわば大域的な再現率を指すことが一般的である．それに対して，ここでの再現率は，文書中の適合箇所のうちの抽出できた割合であり，局所的な再現率や適合箇所の抽出率とも言い換えることが可能である．

また，上記の評価尺度に加えて，手法ごとに作成される検索結果のテキストサイズの計測も行った．

5.4.2　実 験 結 果

各クエリごとに上位 1 件，5 件，10 件における検索性能を計測した結果を表 4 に示す．なお，比較手法は，前述の手順で検索結果を作成する **部分文書検索手法**，再構造化手法によって作成された文書集合に対して同様の手順で検索結果を作成する **再構造化部分文書手法** (再構造化手法)，更に，文書全体の粒度の部分文書のみを抽出して検索結果を作成する**文書検索手法**の，計三つである．なお，部分文書検索手法と再構造化手法はいずれもサブ・コンテンツフィルタを適用する．

その結果，表 4 の通り，部分文書検索手法，再構造化手法ともに，いずれの段階においても文書検索手法と比較して検索精度が高いという結果が得られた．また，再現率においては，文書検索手法の性能が部分文書検索手法と再構造化手法を大きく上回った．F 値においても文書検索手法の性能が高いという結果が得られた．これらの結果より，検索結果の精度を重視するのであれば部分文書検索の枠組みは有用であるといえる．

また，部分文書と再構造化を比較した場合に，いずれの段階においても，検索精度と再現率ともに再構造化手法が高い得点を持った．更に，多重比較を行った結果，有意水準 5% において部分文書検索手法と再構造化手法の検索精度には優位に差があるという結果が得られた．同様に，部分文書手法と文書検索手法，再構造化手法と文書検索手法の検索精度において，有意水準 1% において優位に差があることが判明した．

続いて，検索結果の平均テキストサイズと標準偏差の計測を行った．図 5 の通り，部分文書検索手法及び再構造化手法と比

較して，文書検索手法の検索結果は極めて大きくなった．また，再構造化手法よりも部分文書検索手法にて抽出されたテキストサイズが大きくなった原因として，部分文書検索手法では適切な粒度の部分文書の抽出ができない場合に文書全体が検索結果として提示される場合があったためである．検索結果のテキストサイズが小さいほどユーザの負担は小さいと考えられるため，再構造化手法が最もユーザの負担が少ないといえる．

また，再構造化によって概ね適切な粒度の部分文書が作成されたものの，一部，再構造化によって起こる問題が確認された．それは，文書中の最後に出現した Heading タグの後ろにサブ・コンテンツやフッター情報が付随し，その結果適合箇所でない部分を含めて新たな部分文書の粒度が定義されるという問題である．このような問題を解決するために，サブ・コンテンツとして判定された部分文書は，その先祖部分文書が検索結果として提示する際にも除外することが必要であることを示唆された．

6. おわりに

本稿では，XML 部分文書検索技術の Web 文書への適用へ取り組んだ．代表的な Web 文書のファイルフォーマットである HTML 文書は，XML 文書と比較すると，1) 文書構造が整形されていない，2) 文書内容の論理構造と文書の物理構造が一致していない，3) サブ・コンテンツのように，情報要求を満たさない箇所を多分に含むという特徴を持つ．1) の特徴は既存の HTML 文書整形ツールに解決可能であるために，本稿では 2) と 3) の特徴によって発生する課題の解決に取り組んだ．

それぞれの課題の解決のために，論理構造を抽出する上での手掛かりタグを利用した文書構造の再構造化手法と，サブ・コンテンツから構成される部分文書中には他ページへのハイパーリングが多数出現することを利用してサブ・コンテンツを判別するためのフィルタの提案を行った．

文書の再構造化の評価実験を行った結果，文書構造に新たな粒度を付与することに成功し，サブ・コンテンツフィルタは適切に判別を行うことを示した．また，部分文書検索を Web 文書検索へ適用した結果，検索精度は文書検索を上回った．また，再構造化手法を適用することで，抽出するテキストサイズを抑制しつつ検索精度と再現率も向上した．

今後の課題として，再構造化によって抽出された，文書末尾の情報要求を満たさない箇所を取り除く仕組みが必要である．また，今回用いたクエリは部分文書検索との相性がよいものを優先的に利用したために，今後より多用なクエリに対しての性能評価を行うことや，1CLICK-2 タスクにおいての部分文書検索技術の有用性の評価を行うことなどが考えられる．

謝　辞

本研究の一部は，JSPS 科研費 (特別研究員奨励費，基盤研究 (A)(課題番号:22240005)，基盤研究 (C)(課題番号:23500121)，若手研究 (B) (課題番号:22700248)) の支援による．ここに記して謝意を表す．

文　献

[1] Deng Cai, Shipeng Yu, Ji-Rong Wen, and Wei-Ying Ma. Extracting Content Structure for Web Pages based on Visual Representation. In *Proc. of the 5th ACM APWeb*, pp. 406–417, 2003.

[2] Jaap Kamps, Shlomo Geva, Andrew Trotman, Alan Woodley, and Marijn Koolen. Overview of the INEX 2008 Ad Hoc Track. In *INEX 2008 Workshop Pre-proceedings*, pp. 1–28, 2008.

[3] Jaap Kamps, Shlomo Geva, Andrew Trotman, Alan Woodley, and Marijn Koolen. Overview of the INEX 2008 Ad Hoc Track. In *Formal Proc. of INEX 2008 Workshop*, Vol. 5631 of *LNCS*, 2009.

[4] Marcin Kaszkiel and Justin Zobel. Passage retrieval revisited. In *Proc. of the 20th ACM SIGIR*, pp. 178–185, 1997.

[5] Makoto P. Kato, Matthew Ekstrand-Abueg, Virgil Pavlu, Tetsuya Sakai, Takehiro Yamamoto, and Mayu Iwata. Overview of the NTCIR-10 1CLICK-2 Task. In *Proc. of the 10th NTCIR Conference*, 2013.

[6] Gabriella Kazai, Mounia Lalmas, and Arjen P. de Vries. The Overlap Problem in Content-Oriented XML Retrieval Evaluation. In *Proc. of the 27th ACM SIGIR*, pp. 72–79, 2004.

[7] Atsushi Keyaki, Kenji Hatano, and Jun Miyazaki. Result Reconstruction Approach for More Effective XML Element Search'. *International Journal of Web Information Systems (IJWIS)*, Vol. 7, No. 4, pp. 360–380, 2011.

[8] Atsushi Keyaki, Jun Miyazaki, Kenji Hatano, Goshiro Yamamoto, Takafumi Taketomi, and Hirokazu Kato. Path Expression-based Smoothing of Query Likelihood Model for XML Element Retrieval. In *Proc. of the 1st ACIS International Symposium on Applied Computing and Information Technology*, pp. 296–300, 2013.

[9] Seung-Jin Lim and Yiu-Kai Ng. Converting the Syntactic Structures of Hierarchical Data to Their Semantic Structures. *Information Organization and Databases*, Vol. 579, pp. 343–355, 2000.

[10] Fang Liu, Clement Yu, Weiyi Meng, and Abdur Chowdhury. Effective Keyword search in Relational Databases. In *Proc. of ACM SIGMOD*, 2006.

[11] Wei Liu, Stephen Robertson, and Andrew Macfarlane. Field-Weighted XML Retrieval Based on BM25. In *Formal Proc. of INEX 2005 Workshop*, Vol. 3977 of *LNCS*, 2006.

[12] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schutze. *Introduction to Information Retrieval*, pp. 157–159. Cambridge University Press, 2008.

[13] Satoshi Nakamura. Trustworthiness Analysis of Web Search. *Journal of Japanese Society for Artificial Intelligence*, Vol. 23, No. 6, pp. 767–774, 2008.

[14] Paul Ogilvie and Jamie Callan. Parameter Estimation for a Simple Hierarchical Generative Model for XML Retrieval. In *Formal Proc. of INEX 2005 Workshop*, Vol. 3977 of *LNCS*, 2006.

[15] Benjamin Piwowarski and Patrick Gallinari. A Bayesian Framework for XML Information Retrieval: Searching and Learning with the INEX Collection. *Journal of Information Retrieval*, Vol. 8, No. 4, pp. 655–681, 2005.

[16] 波多野賢治, 絹谷弘子, 吉川正俊, 植村俊亮. XML 文書検索システムにおける文書内容の統計量を利用した検索対象部分文書の決定. 電子情報通信学会論文誌, Vol. J89-D, No. 3, pp. 422–431, 2006.

[17] 欅惇志, 波多野賢治, 宮崎純. 有益な検索結果提示のための部分文書再構成手法の提案. 情報処理学会論文誌：データベース, Vol. 4, No. 1, pp. 1–13, 2010.

[18] 欅惇志, 宮崎純, 波多野賢治, 山本豪志朗, 武富貴史, 加藤博一. 文書の更新を考慮した高精度 XML 部分文書検索手法の提案. 情報処理学会論文誌：データベース, Vol. 6, No. 4, pp. 1–16, 2013.

[19] 徳田隆志, 田島敬史. XML タグの文書構造上の役割に関する自動分類. *DBSJ Journal*, Vol. 8, No. 1, pp. 1–6, 2009.