DEIM Forum 2011 E6-6

XML 部分文書検索における緩和大域的重み付け手法の提案

欅　悼志† 波多野賢治†† 宮崎　純†††

† 同志社大学大学院文化情報学研究科　〒 610–0394 京都府京田辺市多々羅都谷 1–3
†† 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
††† 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
E-mail: †keyaki@ilab.doshisha.ac.jp, ††khatano@mail.doshisha.ac.jp, †††miyazaki@is.naist.jp

あらまし　XML 文書を対象とした情報検索では，文書単位よりも細かな粒度である部分文書を対象とした検索を行うことが可能である．現在利用されている部分文書に対する情報検索技術の多くは，文書検索用の情報検索技術を部分文書検索用に拡張させたものである．その際，大域的重みの算出方法として，1) 各部分文書の文書中の位置を示す際の path 式が同一の部分文書ごとの統計量によって算出，2) 部分文書を囲むタグのタグ名が同一の部分文書ごとの統計量によって算出，の二種類の方針が存在する．前者の場合，複雑な文書構造 (path 式) になるほど，同じ path 式で表される部分文書数が減少するために “大域的” 重み付けにも関わらず少数の部分文書情報のみを利用することになり，また，後者の場合は全く文書構造を考慮していないという問題が存在する．我々はこれらの問題を解決するために，path 式は順序を持つタグの列であると見なし，タグの出現順序や出現頻度に曖昧性を持たせ同一属性を持つ path 式の分類を行うことで，各属性に分類される部分文書数を増加させることを目的とした緩和大域的重み付け手法の提案を行う．評価実験の結果，提案手法を適用することで，検索精度を低下させることなく，属性に含まれる部分文書数が少ない属性の割合を減少させることに成功し，更に，空間計算量が削減されるという効果も得られた．
キーワード　XML 部分文書検索，大域的重み付け手法，path 式の統合

A Method of Relaxation Global Weight for XML Element Search

Atsushi KEYAKI†, Kenji HATANO††, and Jun MIYAZAKI†††

† Graduate School of Culture and Information Science, Doshisha University
1–3 Tatara–Miyakodani, Kyotanabe, Kyoto 610–0394, Japan
†† Faculty of Culture and Information Science, Doshisha University
1–3 Tatara–Miyakodani, Kyotanabe, Kyoto 610–0394, Japan
††† Graduate School of Information Science, Nara Institute of Science and Technology
8916–5 Takayama, Ikoma, Nara 630–0192, Japan
E-mail: †keyaki@ilab.doshisha.ac.jp, ††khatano@mail.doshisha.ac.jp, †††miyazaki@is.naist.jp

1. はじめに

Extensible Markup Language(注1) (XML) はデータ交換のためのマークアップ言語である．World Wide Web (WWW) の発達による Web 文書の増大や，高性能化されたデータベースシステムによって実現される大量データの管理などの影響によって，テキスト情報のみを保持するプレーンテキストではなく構造を持つデータが多数出現している．現実世界においては，必ずしも厳密な構造を持たない半構造データも多数存在するため，自由度の高い記述が実現可能な XML は多くの状況において利用されている．このような背景から，現在，Web 上には膨大な数の XML 文書が存在しており，今後もますます多くの XML 文書が作成されるものと考えられる．膨大な量のデータからユーザの情報要求を満たす情報を取り出すことは非常に困難な作業であるため，XML 文書に対する情報検索技術は重要な技術である．

XML 文書に対する検索では，文書単位よりも細かな粒度，すなわち**部分文書**単位に対して検索を行うことが可能である．本稿では，部分文書を XML 文書中の任意の開始タグと対応する終了タグで囲まれた範囲のテキスト部分を指すこととする．

(注1)：http://www.w3.org/XML/

そのため，一つの XML 文書からタグのペアの個数だけ重複を含む部分文書が取り出せることになる(注2)．このように如何なる粒度の部分文書においても検索結果として提示することが可能であるという特性から，部分文書検索の最大の目標は，情報要求を満たす文書を提示すること以上に，情報要求に合致する箇所そのものをユーザに対して提示することである(注3) [3]．この結果，ユーザは検索結果として提示される文書中から適合箇所を発見するために自ら文書を探索する必要がなくなり，情報検索における労力を大幅に軽減することが可能となる．

このような部分文書を単位とした情報検索では，文書に対する情報検索技術が部分文書検索用に拡張され，利用される場合が多い．現在多用されている部分文書検索技術の大域的重みの計算方法としては，1) 各部分文書の文書中の位置を示す際の path 式が同一の部分文書ごとの統計量によって算出，2) 部分文書を囲むタグのタグ名が同一の部分文書ごとの統計量によって算出，の二種類の方針が存在する．前者の場合，複雑な文書構造 (path 式) になるほど，同じ path 式で表される部分文書数が減少するために，“大域的” 重み付けにも関わらず少数の部分文書情報のみを利用することになるために大域的とはいえないという問題が存在する．また，後者の場合は全く文書構造を考慮していないという問題が存在する．これらの問題を解決するためには，各部分文書の持つ文書構造の情報を利用しつつ，“大域的” な統計量を算出する際に必要な最低限の部分文書群から大域的重みを算出する必要がある．なお，情報検索においてはさまざまな種類の統計量を算出する必要があり，扱うデータ量が大規模化するに連れて処理コストも大きくなる．従って，適切な大域的重み付け手法を提案する上では，低コストで実現可能な手法が望ましい．

そこで我々は，path 式を順序を持つタグの列であると見なし，タグの出現順序や出現頻度の観点から見て類似している path 式は同じ属性を持つと考え，大域的重み計算の際には属性が等しい path 式で表される部分文書群から統計量を算出する．これにより，文書構造を利用しつつ，各属性に含まれる部分文書数が極めて少なくなるという問題を軽減させることを目指す．つまり，path 式の統合アプローチによる緩和大域的重み付け手法の提案を行う．

以降，2. では従来の部分文書情報検索情報検索と関連研究について説明し，3. と 4. では提案手法の説明と有用性を測るための評価実験を行い，5. で本研究のまとめと今後の課題について述べる．

2. 基本的事項と関連研究

本節では，部分文書に関する概要を説明した後に，文書検索用情報検索技術が部分文書検索用情報検索技術として拡張される過程，現在多用されている部分文書用情報検索技術とその大域的重み付け法の定義，構造指定の緩和に関する関連研究について述べる．

(注2)：部分文書とその重複関係に関しては 2.1 にて詳述する．

(注3)：文書検索における検索精度は，多くの場合において検索結果として抽出した文書中の適合文書の割合で測るが，部分文書検索における検索精度とは，抽出したテキストのうちの，適合文書中の適合箇所の割合で測る．

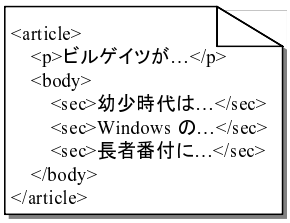

図 1　XML 文書

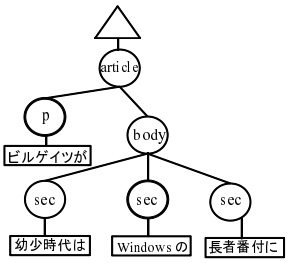

図 2　XML　木

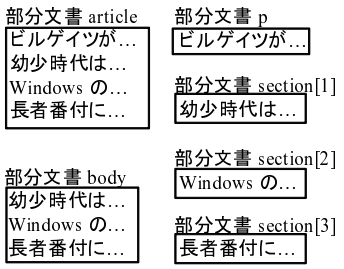

図 3　部 分 文 書

いて述べる．

2.1 部分文書とその重複関係

図 1 は XML 文書の例であり，図 2 は XML 文書を木構造で表現した図である．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，部分文書の説明を容易にすることを目的として，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 木の各ノード名に対応しており，タグの入れ子はノードの親子関係によって表現されている．図 3 の各部分文書は，図 2 の XML 木の各ノード以下に含まれるテキストノードと対応する．つまり，文書全体を表す `article` ノードは子孫に存在するテキストノード全てを持ち，`body` ノードは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保持する．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．

続いて，path 式について具体例を用いて説明する．path 式は根から各部分文書までの経路上に存在するタグを接続した文字列で表現され，部分文書 `p` を表す path 式は `/article/p`，`body` ノードの子の部分文書を表す path 式はいずれも `/article/body/sec` である．

部分文書検索における検索結果提示方法に関して述べる．仮に情報要求を満たす内容が「幼少時代は…」と「Windows の…」，「長者番付に…」であった場合には，検索結果として `article` ノード以下の部分を提示すれば不要な部分 (`p` ノード以下の部分) が含まれ，`body` ノードの子の `sec` ノードのいずれかを提示すればそれ以外の情報要求を満たす部分を取りこぼすという問題が起こる．そのため，この例では `body` ノード以下の部分を提示することが適切である．

2.2 文書検索から部分文書検索への拡張

部分文書を単位とした情報検索では，文書に対する情報検索

技術が部分文書検索用に拡張され，利用される場合が多い．文書検索においては，局所的重み，大域的重み，文書長による正規化，文書集合から算出される各種統計量等によって各索引語に対して重み付けを行う [6]．部分文書検索技術に拡張する際にも同様に，部分文書単位において各種統計量を考慮して索引語に対して重み付けを行うものの，大域的重みの計算方法には工夫が必要である．文書検索において大域的重みを算出する際には，文書は全て等しい属性を持つ対象として扱うために，画一的に重み計算の対象として扱う．しかしながら，部分文書検索では全ての部分文書を同一の属性として扱うのではなく，何らかの基準に沿って分類した上で，分類された各属性ごとに大域的重みを算出することが多い．次節では，現在最も多用されている部分文書検索用情報検索技術の概要と，それぞれの手法においてどのような基準で等しい属性を持つ対象を分類しているのかについての説明を行う．

2.3 主な部分文書検索技術とその大域的重み付け法

代表的な部分文書用スコアリング手法である TF-IPF [2] は，ベクトル空間モデルの文書検索技術である TF-IDF [10] を拡張し，path 式を考慮した大域的重みを用いるスコアリング手法である．TF-IDF は局所的重みとしてある文書中の索引語の出現頻度 (Term Frequency, TF)，大域的重みとして全文書集合中での各索引語ごとの文書頻度の逆数 (Inverse Document Frequency, IDF) の積から算出されるのに対し，TF-IPF はある部分文書に含まれる索引語の出現頻度と，path 式ごとに個別に集計された部分文書頻度の逆数 (Inverse Path Frequency, IPF) の積で算出される．更に索引語数での正規化を行った正規化 TF-IPF [4] などの拡張も存在する．これらの手法においては，完全一致する path 式で表される部分文書は同一の属性を付与されていると考えているため，多少でも path 式が異なれば，それらの部分文書は異なる属性を持つものとして扱われる．従って，構造が深くなればなるほど属性の種類は細分化される．しかしながら，大域的重み付けの趣旨は等しい属性を持つ多くの文書 (部分文書) 集合中での索引語の重みを算出することであるため，各属性に含まれる文書 (部分文書) 数が少なかった場合には適切に大域的な重み付けを行えない可能性がある．そのため，各属性に含まれる部分文書の数が少なくなりすぎないように考慮する必要がある．

同様に，確率モデルに基づいた文書検索用スコアリング手法である Okapi's BM25 [8] を構造化文書検索用に拡張させた BM25F [9] や，部分文書検索用に拡張させた BM25E [5] なども存在する．BM25F ではタグに重みを付与することで，クエリキーワードの出現箇所ごとに出現に対する重みを調整し，効果的な構造化文書検索(注4)を目指している．それに対して，BM25E は TF-IDF から TF-IPF への拡張と同様，部分文書検索用の検索技術へ対応させている．その際，大域的重み付けの計算手法として，全ての部分文書が等しい属性を持つと見なす方針 (Inverse Element Frequency, IEF) と，各部分文書を木構造として見立てた場合のノード名 (タグ名) が等しい部分文書ごとに等しい属性を持つと見なす方針 (Inverse Tag Frequency, ITF) の二つが用いられる [7]．ただし，全ての部分文書が同一の属性を持つと見なすと，同一文書内の部分文書間には重複関係が存在するために，大域的重みを計算する際に重複したテキスト部分が多重に考慮されてしまうことになり，タグを多く含む文書の影響が大きくなるという問題が起こるために大域的重み付け手法として相応しくない．また，タグ名ごとに分類した場合には，タグごとによる部分文書の性質を考慮してはいるものの，その際の文書構造を全く考慮していないという問題が起こる．XML 検索では一般的にキーワードと文書構造の二つを指定することからも，部分文書を分類する際に文書構造を完全に無視することが適切であるとは考えにくく，仮に文書の属性が文書構造の影響を受けるというのであればこれは不適切な処理である．

(注4)：例えば，タイトルや見出しなどに索引語が出現する場合は文書が適合する可能性が高いと見なし，それらのタグの重みを大きくする，などである．

1: /article/sec
2: /article/sec/sec
3: /article/sec/person/sec
4: /article/sec/p/
5: /article/person/sec
6: /article/sec/sec/person
7: /article/person/sec/sec
8: /article/sec/sec/p

図 4　path 式の例

2.4 問合せにおける構造制約の緩和

上記の理由から，path 式の持つ属性が同等であると見なせる範囲内で path 式の統合を行う必要があると考えられる．過去に行われた問合せにおける構造制約緩和に関する研究において，予め path 式ごとに分類した部分文書群を用いて大域的重みや索引語の重みを算出し，その後ユーザによってクエリが入力された段階で，クエリの制約を満たす構造を持つ部分文書の統計量を用いて各索引語の得点を算出している [12]．つまり，前処理の段階では path 式の異なる部分文書は異なる属性を持つ部分文書であると見なして分類しておき，クエリ処理を行う段階で実質的に同等と考えられる構造を集約して統計量を統合するというわけである．しかしながら，これではクエリが投げかけられた段階で初めて同一の属性であるのかどうかの判定を行うことが可能となるために，同一の属性を持つ経路であるかどうかがクエリに依存し，予め同一属性を持つ部分文書の統計量を算出できないために検索に要する時間が増加する，などといった問題が起こる．

3. 提 案 手 法

2. で紹介した各手法の大域的重み付け法の問題点と過去の構造緩和に関する研究の問題点を踏まえて，各属性に含まれる部分文書の数が少なくなりすぎないように考慮する必要がある．また，クエリが投げられる以前の段階で既に属性ごとの分類が

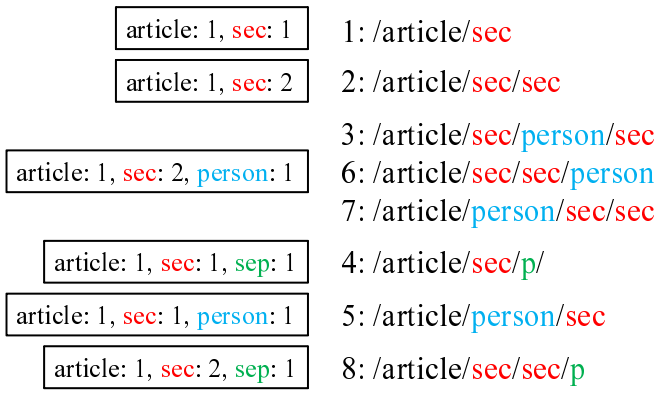

図 5　ICF 法の分類例

行われていることが望ましい．これらの要件を満たすためには同一属性と見なす path 式を統合する必要があると考えられるが，このとき，極力各部分文書の持つ構造情報を加味しつつ類似した path 式同士を統合する必要がある．従って，本稿では path 式統合による緩和大域的重み付け手法を提案する．なお，部分文書検索技術を利用する際には，単に大域的重みだけではなく，局所的重みや部分文書長による正規化を始めとしてさまざまな統計量を算出する必要がある．これらの統計量は，検索対象として扱うデータ量が大規模化するに連れて処理コストも大きくなるため，大域的重みを算出することに多大なるコストを割くことは現実的ではない．つまり，path 式統合手法を提案する上では極力低コストで実現可能な手法であることが期待される．

path 式による分類によって算出される大域的重みでは，path 式の経路上に出現するタグの順序と出現回数 (頻度) を厳密に考慮していると見なすことが可能である．従って，path 式をそのまま利用するのではなく，緩和された構造の指定を分類基準とするのであれば，タグの順序と頻度において曖昧性を持たせることが可能であると考えられる．以降，1) タグの出現順序を緩和させた大域的重み付け法，2) タグの出現回数を緩和させた大域的重み付け法，3) タグの出現順序と出現回数両方を緩和させた大域的重み付け法の計三種類の緩和大域的重み付け法を提案する．

3.1　組合せ分類法

組合せ分類 (Inverse Combination Frequency, ICF) 法は path 式中のタグの出現順序の緩和を行う大域的重み付け手法である．XML 文書中で定義されるタグは，構造を表すタグや強調を表すタグや何らかのメタ属性が付与されたタグなど，さまざまな機能を持つタグが存在する．そのため，各タグの生起が互いに独立である場合も多く存在することが予想される．これはつまり，ある path 式を構成するタグの組合せで，異なる path 式が定義される可能性があるということを意味する．そのような場合においてそれぞれの path 式が異なる属性に分類されることが必ずしも適切ではないと考える．従って，厳密に path 式中に出現するタグの順序を考慮するのではなく，path 式中に出現するタグの種類と出現回数の組合せを考慮することで属性の分類を行う．このような考えのもと，ICF 法では根から葉ノードにかけてのタグの出現する順序は考慮せずに path 式の分類を行う．

ICF 法において等しい属性と分類される path 式の例を示すため，図 4 の path 式が分類された結果を図 5 に示す．分類を行うために，まずは path 式中に現れるタグの種類と出現回数を列挙する．その結果，タグの種類と出現頻度が同一の path であるのは path 式 3, 6, 7 の組み合わせのみであり，これらの path 式は全て，`article` が一回，`sec` が二回，`person` が一回出現する．従って，これらの path 式で表される部分文書は同一の属性を持つして大域的重みを計算する．なお，path 式 1, 2, 4, 5, 8 に関しては各 path 式ごとに統計量が算出される．

3.2　集約的分類法

続いて，タグの出現回数を緩和させた大域的重み付け法である集約的分類 (Inverse Aggregated-path Frequency, IAF) 法について述べる．XML 文書中において，同名のタグが連続して複数回出現することが起こりうる．このような場合に，タグが表す概念そのものは常に固定されているために，同名のタグが連続して出現することで path 式の持つ意味が大幅に変更されるとは考えにくい．従って，IAF 法では path 式中のタグの出現回数に曖昧性を持たせることを考慮する．また，XML 文書の生成ルールを記述する Document Type Definition (DTD) によってはタグ (要素) の包含関係が定義されている場合がある．そのような場合に，経路のタグの情報を全く顧みないということは必ずしも適切であるとは考えられない．従って，IAF 法では，path 式中に出現するタグの順序は損失なく考慮しつつ，曖昧性を許容した path 式の分類を行う．つまり，同名のタグが連続して複数回出現する場合にそれらのタグを集約し，一つの経路に纏める．

先ほどと同様に図 4 の path 式を分類した結果を図 6 に示す．まず，各 path 式に対して，連続して出現する同名のタグを集約する．該当箇所は，path 式 2 の `sec`, path 式 6 の `sec`, path 式 7 の `sec`, path 式 8 の `sec` である．その結果，等しい属性として扱われる path 式の組み合わせは，path 式 1 と 2，path 式 4 と 8, path 式 5 と 7，の三つである．path 式 1, 2 は `article` タグが一度以上出現した後に `sec` タグが一度以上出現し，同様に path 式 4, 8 は `article` タグが一度以上出現した後に `sec` タグが一度以上出現し，更に `p` タグが一度以上出現する．そして，path 式 5, 7 は `article` タグが一度以上出現した後に `person` タグが一度以上出現し，`sec` が一度以上出現する．なお，path 式 3, 6 で表される path 式はそれぞれの path 式で統計量を算出される．

3.3　集合的分類法

上記の ICF, IAF はそれぞれ，path 式中に出現するタグの出現順序もしくは出現回数のいずれかのみを緩和させた大域的重み計算手法であった．それらに対して集合的分類 (Inverse Set Frequency, ISF) 法では，これら二つの観点両方による曖昧性を持たせるために，タグの出現する順番や回数などは一切考慮せずに path 式に含まれるタグからなる集合が等しい部分文書を同一の属性を持つと判断する．

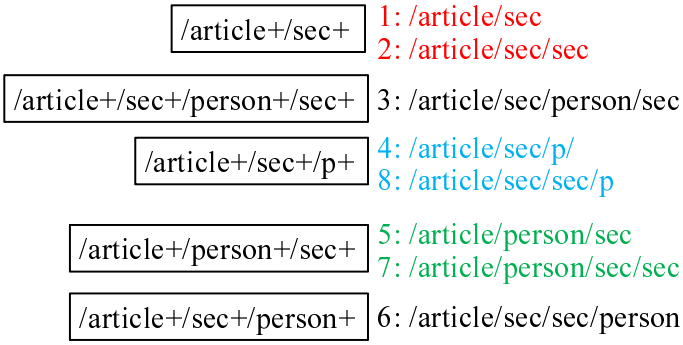

図 6　IAF 法の分類例

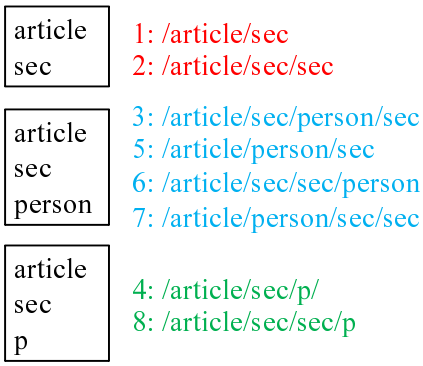

図 7　ISF 法の分類例

これまでと同様に図 4 を ISF の観点によって分類した結果を図 7 に示すと，path 式中に `article` と `sec` タグが含まれる path 式は，path 式 1 と 2 である．また，path 式中に `article`, `sec`, `person` タグが含まれる path 式は，path 式 3, 5, 6, 7 であり，path 式中に `article`, `sec`, `p` タグが含まれる path 式は 4 と 8 である．これにより，タグ間に親子関係の制約や，同一文書中の出現回数の制約が存在しない path 式を効率的に纏めることが可能となる．

4. 評価実験

本節では，提案手法を適用することでどのように path 式が統合されるのか，各属性に分類される部分文書の数はどのように分布するのかを調査する．更に，提案した大域的重み付け手法によって検索精度にどのような影響を及ぼすのか確認するために行った評価実験について述べる．

4.1 実験用テストコレクション

本節の実験には INEX 2008 テストコレクション [3] と INEX 2010 テストコレクション [1] を使用した．このテストコレクションは三つの要素から構成されており，一つ目が INEX document colleciton，二つ目が INEX topics，そして三つ目が INEX relevant assessments である．

INEX document collection は英語版の Wikipedia コーパスである XML 文書の集合から構成されている．INEX 2008 テストコレクションは 2006 年初期に収集された約 660,000 個の XML 文書から構成され，INEX 2010 テストコレクションは 2008 年後期に収集された約 2,670,000 個の XML 文書から構成されている．

表 1　path 式の統合 (INEX 2008 テストコレクション)

手法	属性数 (個)	統合前との比較 (%)
ICT 法	56,369	85
IAF 法	32,421	49
ISF 法	16,007	24
IPF 法	66,210	100
ITF 法	1,257	1.9

表 2　path 式の統合 (INEX 2010 テストコレクション)

手法	属性数 (個)	統合前との比較 (%)
ICF 法	22,743,778	.97
IAF 法	23,383,388	.99
ISF 法	19,587,224	.83
IPF 法	23,502,448	1.0
ITF 法	22,110	.000094

クエリ集合である INEX 2008 topics は合計 68 個のクエリ集合から構成されており，4.4 の検索精度に関する評価実験では全てのクエリを用いた．INEX topics に含まれるクエリは二種類存在し，一方はキーワードのみを指定する CO (Content only) クエリであり，もう一方はキーワードと構造を指定する CAS (Content and Structure) クエリである．また，これらのクエリは全て NEXI (Narrowed-Extended XPath I) [11] クエリで表現される．

INEX relevance assessments は XML 部分文書検索用の評価ツールである．XML 検索エンジンは INEX relevance assessments に対して，非重複リストを提出することで，さまざまな評価尺度による検索精度を評価することができる．

4.2 path 式統合に関する実験

提案手法を適用することで base line の手法である IPF と比較してどの程度 path 式が統合されたのか計測を行った結果を表 1 と表 2 に示す．

表 1 より，INEX 2008 テストコレクションを用いた場合に三種類の提案手法全てにおいて属性数の削減に成功した．その際，タグの出現順序に曖昧性を持たせる ICT 法よりも，タグの出現回数に曖昧性を持たせる IAF 法を用いることで，より効果的に path 式を統合することが可能であるという結果が得られた．また，タグの出現回数とタグの出現頻度両方に曖昧性を持たせた ISF 法では大幅に path 式を統合可能であり，本来の 1/4 程度まで属性数が減少した．

それに対して，表 2 に示す通り，INEX 2010 テストコレクションを用いた実験ではほとんど属性数を削減することができなかった．特に IAF 法の属性数は IPF 法とほとんど変わらなかった．また，ICF 法，IAF 法いずれも削減率が非常に低かったものの ISF 法において多少効果が認められた理由として，今回我々が提案した緩和方法が不十分である可能性が示唆された．特に，IAF 法で十分に頻度を緩和できているのかを考慮する必要がある．

このような結果が得られた理由として，INEX 2010 テストコレクションに含まれるタグの種類は INEX 2008 テストコレク

表 3　INEX 2008 テストコレクションにおける各種属性に含まれる部分文書数の個数と属性の割合 (%)

部分文書数 (個)	IPF 法	ITF 法	ICN 法	IAF 法	ISF 法
1	57	58	53	42	37
2	13	14	14	16	13
3	6.2	5.5	6.7	8.0	6.3
4	4.1	2.6	4.2	5.7	5.2
5	2.7	1.8	2.8	3.1	3.6
6	2.1	0.80	2.2	2.7	3.2
7	1.4	1.0	1.7	1.7	1.9
8	1.4	0.40	1.4	1.9	1.7
9	1.1	1.0	1.2	1.2	1.3
10	9.0	0.60	1.0	1.1	1.0
11〜	10	14	12	17	25

表 4　INEX 2010 テストコレクションにおける各種属性に含まれる部分文書数の個数と属性の割合 (%))

部分文書数 (個)	IPF 法	ITF 法	ICN 法	IAF 法	ISF 法
1	65	61	66	65	65
2	13	14	11	13	12
3	4.7	5.3	5.0	4.8	5.2
4	2.8	3.1	2.9	2.8	3.1
5	1.7	1.8	1.8	1.7	2.0
6	1.3	1.6	1.5	1.3	1.7
7	1.5	1.2	1.3	1.5	0.84
8	1.1	1.2	0.88	1.1	0.70
9	0.61	0.52	0.59	0.60	0.33
10	0.57	0.78	0.59	0.6	1.0
11 〜	8.1	10	8.1	8.1	8.6

ションに含まれるタグの種類に対して大幅に増大したためであると考えられる．その結果 path 式を構成するタグの組合せも増大し，いずれの手法においても効果的に path 式を統合することができなかったということである．従って，我々は提案手法とは異なる緩和条件を考慮しなければならず，例えば，path 式のうち考慮するタグの選別を行うことなどが考えられる．今回我々が提案した手法いずれにおいてもタグの種類については緩和を行わなったが，提案手法は属性統合において不十分であるのであれば，適宜緩和を行うという方向性も視野に入れるべきである．

このような結果が得られた原因について更に深く調査を行うために次節の実験を行った．

4.3　各属性に含まれる部分文書数に関する実験

4.2 の実験によって，提案手法を用いることで，テストコレクションによって path 式の統合の効率が大きく異なるという結果が得られた．続いて，提案手法によって統合された属性を利用することで，実際に属性に含まれる部分文書の個数が極端に少ない属性割合を抑えることが可能であるのかどうかを確認するため，各属性に含まれる部分文書の個数にどのような変化が生じたのかを計測する．表 3 と表 4 はそれぞれ，INEX 2008 テストコレクションと INEX 2010 テストコレクションにおける各属性に含まれる部分文書の個数と，その個数の部分文書を含む属性の割合を示した図である．

表 3 は，各属性に含まれる部分文書の個数と，その個数の部分文書を含む属性の割合を示した表である．従来手法の IPF 法，ITF 法と比較して提案手法はいずれも部分文書数が 1 個のみ存在する属性の割合を減らすことができた．また，部分文書数が 11 個以上存在する属性の割合も IPF 法と比較して増加しているという結果が得られた．これにより，タグの出現順序や出現回数に曖昧性を持たせることで，属性に含まれる部分文書数が著しく少ない属性の割合を削減することができたと結論付けることができる．このとき，部分文書数が 1 個のみ存在する属性の割合が最も低い結果となったのは ISF 法であり，部分文書数が 11 個以上の属性の割合も最も高くなった．これらの結果から，属性に含まれる部分文書の個数が少ない属性の割合を減らす上では ISF 法が最も効果的であるという結果が得られた．なお，適切に大域的重み付けを算出するために必要な部分文書数がどの程度であるのかについては議論の余地があるものの，少なくとも path 式の統合を行うことで，属性に含まれる部分文書数が極端に少ない属性を減少させる上で一定の効果は認められた．

また，ITF 法に着目した場合に，属性が含む部分文書数が少ない属性は数多く見られた．タグは合計で 495 種類存在しているが，そのうちの 289 種類 (58%) は全文書群のうちで一度しか出現しておらず，ごく一部のタグが繰り返し多用されているということである．つまり，今後更に属性に含まれる部分文書数が著しく少ない属性の割合を軽減させるためには，上記のごく僅かな出現回数のタグに着目することが必要であることが示唆された．

続いて，表 4 を見ると，4.2 の実験結果からも予想できた通り，提案手法と従来手法ではほとんど差が生じなかった．従って，INEX 2010 テストコレクションにおいても属性数を削減するためには何らかの手法を提案する必要がある．

更に，提案手法を用いて path 式の統合を行うことで，従来手法の IPF 法と比較して大域的重みを算出する際の空間計算量を削減することが可能である．大域的重みの算出に必要な領域としては，各属性に分類される部分文書数と，各属性の各索引語に対しての文書頻度である．今，検索対象文書全体に存在するユニークな path 式の数を n，各部分文書に含まれる索引語の平均を t とすると，IPF 法による大域的重み算出のために必要な空間は $n(1+t)$ となる．このとき，提案手法によって統合された後の path 式の割合を $p(0 \leqq p \leqq 1)$ とすると，提案手法による大域的重み算出のために必要な空間は $pn(1+t)$ となるため，結局 $n(1-p)(1+t)$ 分の空間を削減可能である．

今回我々が実験に用いた Wikipedia データはある時点のスナップショットを利用しているが，Wikipedia のように更新が頻繁に行われる文書群に対して Web 検索を行う場合には，検索精度を保つためには各種統計量を頻繁に再計算する必要があり，大域的重みに関しても同様である．その際，検索対象となる文書群が大規模であった場合には各種統計量の算出のための必要な空間も大きくなるため，空間計算量を削減することで得られる利点は大きい．

表 5　大域的重み付け手法の検索精度への影響

手法	iP[.01]	MAiP
ICT 法	.6169	.1713
IAF 法	.6178	.1724
ISF 法	.6166	.1716
IPF 法	.6146	.1723
IEF 法	.6107	.1321
ITF 法	.6135	.1719
大域的重みなし	.2364	.04689

4.4　検索精度に関する評価実験

4.4.1　評 価 尺 度

Web 検索システムユーザに対する調査結果において，ユーザは提示された検索結果のうち上位の数件しか確認しないと報告されている [13] ことからも，高精度情報検索技術において最も重要視すべき評価項目としては，検索精度上位における検索精度である．同様の考えのもと，INEX では再現率が 1% 点における補間適合率である iP[.01] を公式な尺度としており，本稿においても iP[.01] を検索手法の評価指標とする．また，INEX project で利用されるその他の評価尺度としては Mean Average interpolated Precision (MAiP) が存在する．MAiP は複数の再現率点における精度の平均から求められ，INEX project においては 101 個の再現率から計算される．本稿では参考のために MAiP も記載することとした．

4.4.2　実 験 方 法

提案した三つの緩和大域的重み付け法によって算出された大域的重みを用いて，検索精度を調査する．このとき，従来手法に対する提案手法の優位性確認のために，従来手法による重み付けによる大域的重み付けを用いた評価実験も行う．なお，各大域的重み付け法を適用させる部分文書検索技術としては BM25E を利用する(注5)．ただし，INEX 2010 テストコレクションにおいては提案手法の効果は認められなかったために，ここでは検索精度の評価実験は行わない．

4.4.3　大域的重み付けによる検索精度への影響

評価実験の結果を表 5 に示す．提案手法である IAF 法, ICT 法, ISF 法の検索精度が IPF 法よりも高精度であったことから，大域的重み付けの方法において path 式ごとに統計量を算出するよりも緩和的に同一属性の構造を拡張させることで検索精度が向上するということが判明した．また，提案手法は IEF 法, ITF 法よりも高い精度を示したということからも，全く文書構造を考慮しないという大域的重み付け法も同様に不適切であるということが判明した．更に，大域的重みを考慮せずに検索精度を計測した場合には検索精度は急激に低下したことより，高精度部分文書検索のためには適切に大域的重みを算出する必要があるということが明らかになった．

提案手法間の検索精度の比較を行った場合に，IAF 法, ICT 法, ISF 法の順に高い検索精度を示したことから，タグの出現順序を翻すことよりもタグの重複を無視することで，より効果的な大域的重み付けを行うことが判明した．また，タグの出現順序と出現回数両方を緩和した場合に，それぞれの緩和を施した場合に比べて検索精度が下がったことから，緩和された構造の自由度が高すぎては逆効果に陥るということが示唆された．

4.2〜4.4 の結果より，統合の効率を優先するのであれば path 式中に出現するタグの種類のみに着目した ISF 法，検索精度を優先するのであれば path 式中で連続で出現するタグの集約を行う IAF 法が提案手法として適切であると判明した．

(注5)：部分文書 i の索引語 j の重みは $\frac{(k_1+1)tf_{i,j}}{k_1((1-b)+b\frac{el}{avel})+tf_{i,j}} \log \frac{N-df_i+0.5}{df_i+0.5}$ で算出される [5]．パラメータに関しては $k_1 = 2.5, b = 0.85$ を設定した．

5.　お わ り に

本稿では，部分文書検索における大域的重み計算のための，path 式の分類手法の提案を行った．その際，path 式に曖昧性を持たせた分類を行うことで，実質的に同等と見なせる path 式ごとに分類を行った．これにより，少数の部分文書の情報から大域的重み付けのための統計量の計算が行われることが軽減されることを目指した．

提案手法を適用することで，INEX 2008 テストコレクションでは属性に含まれる部分文書の個数が極端に少ない属性を減少させることにに成功し，その副作用として，空間計算量が削減されるという効果も得られた．しかしながらその一方で，INEX 2010 テストコレクションでは提案手法の効果が認められなかった．

また，評価実験の結果，緩和された構造による分類を行うことで検索精度が向上するという知見が得られたが，その一方で，緩和しすぎると却って効果が薄れるという結果が得られた．

今後の課題として，全文書群中において低頻度出現タグをどのような扱うのか考慮する必要があり，また，INEX 2010 テストコレクションにおいて提案手法が上手く働かなかった原因についても引き続き調査を行う予定である．

謝辞　本研究の一部は，文部科学省科学研究費補助金特定領域研究 (課題番号: 21013035)，日本学術振興会科学研究費補助金若手研究 (B) (課題番号: 22700248)，日本学術振興会科学研究費補助金基盤研究 (A) (課題番号:22240005) によるものである．ここに記して謝意を表す．

文　　献

[1] Paavo Arvola, Shlomo Geva, Jaap Kamps, Ralf Schenkel, Andrew Trotman, and Johanna Vainio. Overview of the INEX 2010 Ad Hoc Track. In *INEX 2010 Workshop Pre-proceedings*, pages 11–40, December 2010.

[2] Torsten Grabs and Hans-Jörg Schek. PowerDB-XML: A Platform for Data-Centric and Document-Centric XML Processing. In *Proceedings of the First International XML Database Symposium*, volume 2824 of *Lecture Notes on Computer Science*, pages 100–117. Springer Berlin, September 2003.

[3] Jaap Kamps, Shlomo Geva, Andrew Trotman, Alan Woodley, and Marijn Koolen. Overview of the INEX 2008 Ad Hoc Track. In *Advances in Focused Retrieval*, volume 5631 of *Lecture Notes on Computer Science*, pages 1–28. Springer Berlin, September 2008.

[4] Fang Liu, Clement Yu, Weiyi Meng, and Abdur Chowdhury. Effective Keyword search in Relational Databases. In *Proceedings of the 2006 ACM SIGMOD International Conference on Management of Data*, pages 563–574. ACM, June 2006.

[5] Wei Liu, Stephen Robertson, and Andrew Macfarlane. Field-Weighted XML Retrieval Based on BM25. In *Advances in XML Information Retrieval and Evaluation*, volume 3977 of *Lecture Notes on Computer Science*, pages 161–171. Springer Berlin, June 2006.

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

[7] Benjamin Piwowarski and Patrick Gallinari. A Bayesian Framework for XML Information Retrieval: Searching and Learning with the INEX Collection. *Journal of Information Retrieval*, 8(4):655–681, December 2005.

[8] Stephen Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. Okapi at TREC-3. *The Third Text REtrieval Conference (TREC-3)*, pages 109–126, 1995.

[9] Stephen Robertson, Hugo Zaragoza, and Michael Taylor. Simple BM25 Extension to Multiple Weighted Fields. In *Proceedings of the 13 ACM International Conference on Information and Knowledge Management*, pages 42–49, November 2004.

[10] Gerard Salton and Christopher Buckley. Term-Weighting Approaches in Automatic Text Retrieval. *Journal of Information Processing and Management*, 24(5):513–523, January 1988.

[11] Andrew Trotman and Börkur Sigurbjörnsson. Narrowed Extended XPath I (NEXI). In *Advances in XML Information Retrieval*, pages 16–40, May 2005.

[12] 波多野 賢治, シーハム アメルヤヒア, and ディベッシュ スリバスタバ. XML 情報検索における構造問合せを利用した部分文書スコアリング. In **電子情報通信学会技術研究報告**, number 254, pages 13–18, October 2007. DE2007-117.

[13] 中村 聡史. 情報信頼に対する信頼性調査および結果. **人工知能学会誌**, 23(6):767–774, November 2008.