Web コンテンツの相当関係の発見とその応用

平野　雄也† 　馬　　強†† 　吉川　正俊††

† 京都大学工学部情報学科　〒 606–8501 京都市左京区吉田本町
†† 京都大学大学院情報学研究科　〒 606–8501 京都市左京区吉田本町
E-mail: †y.hirano@db.soc.i.kyoto-u.ac.jp, ††{qiang,yoshikawa}@i.kyoto-u.ac.jp

あらまし　大学や企業などの組織では，各部局の Web サイトにより情報の公開が行われることが多い．このような Web サイト間では，例えば教員情報や組織情報などのように，本来は同じ情報を表すべき部分について記述の不一致がよく起こる．本研究では，Web ページの内容を木構造によりモデル化して部分木の比較を行い，これにリンク構造および更新時間を考慮することで相当関係にある Web コンテンツの組を発見する手法を提案する．さらに，コンテンツの相当関係に基づく Web ページ間の記述の不一致を検出する．実際に大学の Web サイトを対象に評価実験を行い，本手法の有効性を検証する．

キーワード　Web マイニング，相当関係，情報抽出，大学の Web サイト

1. はじめに

近年，大学や企業などの組織では，各部局の Web サイトによって情報の公開が行われることが多い．例えば京都大学の場合は，大学，学部，学科，大学院研究科，専攻，研究室など，多くのサイトが互いに関連し合い，情報の公開を行っている．こうした複数のサイト間では，教員情報，研究テーマ，発表論文などが複数のページに記述され，サイトの管理者やページの更新時期が異なるために記述の不一致が起こりやすいという問題がある．このような不一致を修正するには，膨大なコストがかかる．特に，組織が大規模になるほどサイトも大規模になり，不一致の記述箇所を発見するには多大な労力を必要とする．

そこで本研究では，複数の Web ページ間において同じような情報を記述した部分を特定する手法を開発し，不一致となる箇所の発見を支援することを目的とする．本論文では，データやオブジェクトの構成要素とその対応関係が同一であると思われる Web コンテンツ間において成り立つ関係を，**相当関係**と呼ぶ．相当関係にあるコンテンツは共通のデータベースをもとに生成されるが，各ページによって参照されるデータやその表現形式が異なることがある．コンテンツ間の相当関係は，例えば図 1 のように，ある研究室の教員情報を表すコンテンツにおいて成り立ち，その部分において記述は一致するはずである．しかし，コンテンツ間に相当関係は成り立つのだが，本来は同じ情報を表すべき部分について記述の不一致が起きている．本研究では，このような記述の不一致を考慮して，相当関係にある Web コンテンツを発見する手法を提案する．

相当関係発見の流れは図 2 のようになる．まず，Web ページのコンテンツが階層構造を成していることに注目し，ページ内の情報をモデル化した木構造（以下，データ木と呼ぶ）を構築する．データ木の構築方法は，既存の情報抽出に関する研究 [1] で提案される手法を利用する．データ木はページ中に現れる構造化データを階層的に表現したものとなる．図 2 左を含んでいるような Web ページに対してデータ木を構築すると，図 2 中央で表されるデータ木の一部ができる．本論文では，構築されたデータ木の比較をボトムアップ的に行うことで，相当関係が成り立つ部分を発見する．

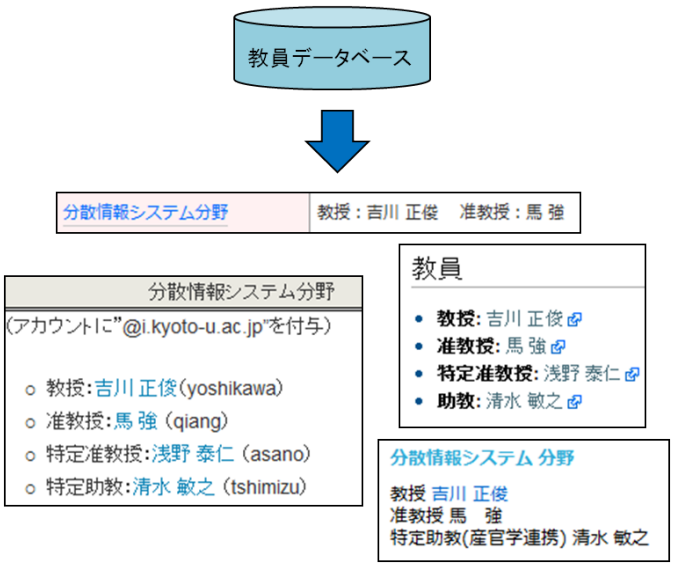

図 1　異なるページ間で相当関係にあるコンテンツの例

そして，構築されたデータ木の集合の中から，相当関係にありそうな部分について評価を行う．相当関係の評価スコア（以下，相当スコアと呼ぶ）算出には，部分木の類似性に加えて，リンク構造と更新時間を利用する．これは，ハイパーリンクの関係にある組織のサイトのページ間ではコンテンツの相当関係が成り立ちやすく，また，更新時間の差によって記述の不一致が起きやすいことを考慮する必要があるという考えに基づくものである．リンク構造および更新時間を利用することで，部分木の類似性があまりない場合にも相当関係を発見できると考える．

相当関係にある Web コンテンツの組を発見することで，異なるページ間での記述の不一致箇所を検出することが可能にな

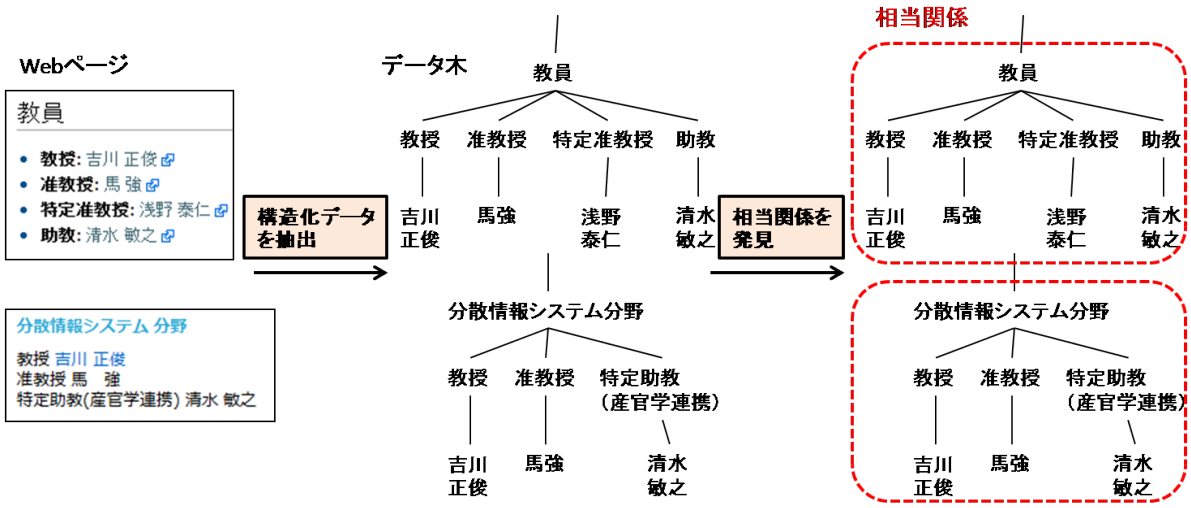

図 2　相当関係発見の流れ

る．また，相当関係にあるコンテンツの組が事前にわかっていれば，一つのコンテンツに対して変更が加えられたときに，それと相当関係にあるコンテンツに対しても変更を行う必要があることを通知するサービスも考えられる．

本論文の構成は以下の通りである．第 2 章では，関連研究について紹介する．第 3 章では，既存の情報抽出手法を利用して構築されたデータ木の部分に対して相当関係が成り立つかどうかを判定する手法を提案する．さらに，構築されたデータ木においてどの部分に相当関係があるのかを発見する手法を提案する．第 4 章では，提案手法の有効性を検証するために，大学の Web サイトを対象に評価実験を行う．第 5 章では，本研究の応用として考えられるアプリケーションシステムについて提案する．最後に第 6 章では，本論文をまとめ，今後の課題について述べる．

2. 関 連 研 究

本研究の関連研究として，Web コンテンツ間において成り立つ関係を発見する研究について第 2.1 章で，Web ページからの情報抽出の研究について第 2.2 章で述べる．

2.1 Web コンテンツ間の関係発見

本研究では，Web コンテンツの相当関係を対象にしているが，包含関係を対象にした関係の発見を行う研究がある [2] [3]．Web ページにおいて成り立つ包含関係として例えば，研究室の各構成員の発表論文一覧は，研究室の発表論文一覧のサブセットである，というようなものである．

澤ら [2] は，HTML 文書中の単純な繰り返し構造（リストなど）に注目し，Web ページのコンテンツを木構造によりモデル化し，包含関係を発見する手法を提案している．木構造において，ある深さに存在するノード集合を比較し，共通のテキスト集合を求める．ルートから共通ノードまでのパスを求め，パスから求められるノード間において完全に包含関係が成り立つかどうかを判定している．

高橋ら [3] は，Web ページの要素が階層構造をなしていることに注目し，Web ページ要素間の包含関係を発見する手法を提案している．ページに含まれるすべての要素の組み合わせに対して包含関係を計算しているため，スコアリングのルールを定義することで，より重要な包含関係の発見を行っている．

本研究では，Web ページから構造化データを抽出して構築されたデータ木から相当関係が成り立つ部分を発見する手法を提案している．内容，構造および更新時間も考慮することで，表現の差に留まらず，コンテンツ間に違いがある場合にも相当関係を発見できる．また，包含関係は 2 つの Web コンテンツ間に成り立つ関係であるが，相当関係は複数の Web コンテンツの組に対して成り立つ関係であるため，複数の Web コンテンツの組を発見する必要がある．

2.2 情 報 抽 出

Web ページの構造に着目して必要なデータを抽出する研究はこれまでに多く行われてきた [4]．データを抽出することで，商品やサービスの比較，情報統合やメタ検索エンジンなどに応用可能となる．情報抽出の手法には，大きく分けて 2 つあり，教師あり学習を用いたものと教師なし学習を用いたものがある．

Muslea ら [1] は，教師ありの機械学習を用いて抽出ルールを生成する手法を提案している．ページとページ内の抽出対象の集合を与えることで，そのページから自動的に抽出ルールを学習・生成する．抽出対象となるデータは，EC 木（embedded catalog tree）と呼ばれる木構造に基づいて抽出される．EC 木の構造は，あらかじめ指定しておく必要があるが，これは例えば図 2 中央のようなものになる．いくつかの EC 木を用意しておくことで，各ページに合った EC 木によって抽出が行われる．抽出ルールは，抽出対象となるデータの前後のパターンを表現したものであり，そのパターンは HTML タグや文字列からなる．

教師あり学習による抽出ルール生成は，人手によるラベリングを行うためメンテナンスコスト・時間がかかり，大規模なサイトには適応できないという問題がある．そのため，教師なし学習による情報抽出手法の研究も多く行われている．

Zhai ら [5] は，Web ページに対して HTML タグツリーまたは DOM ツリーを構築し，木を解析することでデータレコード

を特定する．類似するデータレコードは連続的に配置されるため，繰り返しのパターンから発見可能である．そして，データレコードを含む木を整列させることで，同じデータ項目ごとに整理して表の形で抽出する．

Jindal ら [6] は，リスト構造を含む DOM ツリーに対しても適用可能な木のマッチング手法を提案している．より低いレベルにあるリストを先に見つけることができるため，入れ子にも対応できる．ツリーマッチングの手法により，HTML タグの繰り返しのパターンを発見することで，データレコードの発見とデータの抽出を同時に行うことができる．

本研究では，関連研究 [1] で提案される手法に基づき，Web ページの構造化データを木構造として抽出し，データ木を比較することによって相当関係を発見する．

3. 提 案 手 法

相当関係にある Web コンテンツの組を発見するために，Web コンテンツを木構造によりモデル化し，そのデータ木の比較を行うことで，相当関係にある部分を発見する．二つのデータ木間の相当スコア算出方法ついて第 3.1 章で，データ木間で比較を行う部分を決定する手法について第 3.2 章で述べる．

3.1 相当スコア算出

二つのデータ木間において相当関係が成り立つかどうかを，以下に述べる三つの指標（データ木の類似，リンク構造，更新時間）を利用して判定する．これらの指標を組み合わせることで，二つのデータ木間における相当スコアを計算する．相当スコアがある閾値以上であるときに，相当関係であるとみなす．第 3.1 章では，比較を行う部分はすでに与えられたものとして，相当スコアを計算し，相当関係であるかどうかを判定する方法について述べる．

3.1.1 データ木の類似度

第 3.2 章で述べるが，相当関係は木の葉から探索し，順次その部分を拡大していくことで，最大となる相当関係部分を発見していく．相当関係部分を拡大するごとに，相当関係であるかどうかの判定を行う．それゆえ，ある部分の相当関係を求める際には，その部分木における相当関係にある部分はすでに求まっているため，その結果を利用できる．そこで，二つのデータ木の類似度は，そのデータ木の部分木における相当関係のペアが求まっているという条件のもとで計算を行う．

データ木の類似度の計算は，木の編集距離 [7] [8] に基づき計算する．木の編集距離とは，二つの木を同じ木に変換するための編集操作の最小のコストである．編集操作には，ノードの挿入，削除，置換の三つがある．木の編集距離が小さいほど，木の類似度は大きくなる．二つのデータ木の編集距離を求めるために，木の根ノードを親としたときの部分木に対して，相当関係にあるペアを見つける．一方の部分木に対して相当関係にある部分木が，もう一方に複数存在する場合には，相当スコアの合計が最大（木編集距離の合計が最小）となるような部分木のペアの組み合わせを選ぶ．そして，それ以外の相当関係とされていない（新たに拡大した）部分についても，同様に木編集距離を求める．部分木間で一致するノードがなければ，削除するコストを木編集距離とみなす．

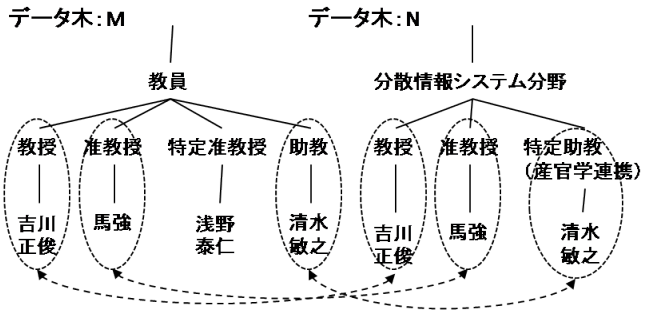

図 3　部分木において相当関係が成立する例

構築したデータ木の特性上，木の葉ノードにおけるマッチングが最重要であると考える．そこで，葉ノードには重みをつけ，葉ノードに対する編集操作のコストは通常のノードに対して 2 倍とする．データ木の編集距離は，相当関係にある部分木のペアの組と，それ以外の部分に分けて考えることができる．二つのデータ木 T_A と T_B の編集距離 $TreeEditDistance(T_A, T_B)$，および木の類似度 $TreeSimilarity(t_A, t_B)$ を以下のように定義する．

$$TreeEditDistance(T_A, T_B) = \sum_{(t_i,t_j)\in eq(T_A,T_B)} TreeEditDistance(t_i, t_j) + TreeEditDistance(T'_A, T'_B)$$

ここで，

- $(t_i, t_j) \in eq(T_A, T_B)$ は，T_A，T_B において相当関係にある部分木のペア (t_i, t_j) を表す
- T'_A，T'_B は，それぞれ T_A，T_B から相当関係にある部分木ペアを除いたものを表す

とする．

$$TreeSimilarity(T_A, T_B) = 1 - \frac{TreeEditDistance(T_A, T_B)}{Nodes(T_A) + Nodes(T_B) - MatchNodes(T_A, T_B)} \quad (1)$$

ここで，

- $Nodes(T_A)$，$Nodes(T_B)$ は，それぞれ T_A，T_B のノード数（葉ノードの重みを考慮したもの）を表す
- $MatchNodes(T_A, T_B)$ は，T_A，T_B において一致するノード数（葉ノードの重みを考慮したもの）を表す

とする．

例えば，図 3 で表されるようなデータ木 M，N の類似度を次のように計算する．破線部分のペアに相当関係が成り立つとする．

- 相当関係にある部分木のペアは 3 組存在する．そのうち，二つのペアの木編集距離は 0 であり，一つのペアの木編集距離は 2 である．よって，相当関係にある部分木のペアの組における木編集距離の和は $0 + 2 = 2$ となる．

• それ以外（相当関係とされていない）部分の木編集距離を考える．一致するノードは存在しないため，これら 4 個のノード（葉ノード 1 個を含む）を削除するコストを木編集距離とみなすと，その値は 5 となる．

以上の議論から，木編集距離は $2+5=7$ と求まる．Mのノード数は 9（葉ノード数は 4）であり，Nのノード数は 7（葉ノード数は 3）である．一致するノード数は 5（葉ノード数は 3）である．よって，MとNの類似度 $TreeSimilarity(M,N)$ は

$$TreeSimilarity(M,N) = 1 - \frac{7}{13+10-8} = 0.53$$

と求めることができる．

3.1.2 リンク構造

二つの Web コンテンツが相当スコアを算出するために，3.1.1 章ではページ内に出現するテキスト情報を木構造にモデル化したものに対して，データ木の類似度を計算した．そのほか Web ページから得られる情報として，ハイパーリンクがある．組織の Web サイト構成は，互いに関連する部局の Web サイト（ページ）がハイパーリンクで参照されることが多いため，ハイパーリンクから Web ページの関連性を求めることができると考える．そして，関連性のある Web ページ間では，相当関係にあるコンテンツを含んでいる可能性が高い．こうした考えに基づき，相当スコアの計算には，リンク構造も利用する．二つの Web ページ A，B 間のハイパーリンクによる関連度（以下，リンク関連度）を，ページ A，B をそれぞれ含むサイト s_A，s_B 中のページ間に存在するハイパーリンクを利用して求める．A，B が同じサイト中のページであれば，リンク関連度を 1 とする．そうでなければ，二つのページ A，B のリンク関連度 $LinkRelevanceDegree(A,B)$ を次のように定義する．

$$LinkRelevanceDegree(A,B) = 1 - \frac{1}{1+(\sum_{j,(i,j)\in(s_A,s_B)} l(i,j) + \sum_{i,(j,i)\in(s_B,s_A)} l(j,i))} \quad (2)$$

ここで，$l(i,j)$ はページ i からページ j へのハイパーリンクがあれば 1 とする．サイト間でハイパーリンクによって参照されるページ数が多いほど，(2) 式で表されるリンク関連度は大きくなる．そのような条件を満たす式であれば，他の式によってもリンク関連度を定義できる．なお，Web サイトに存在する外部サイトへのハイパーリンクは，Website Explorer(注1) というツールを使用して，抽出を行った．

3.1.3 更新時間

Web ページ間で記述の不一致が起こる要因の一つとして，更新時間の違いがある．例えば，同じような情報を記述した二つのページのうち，一方の更新時間が 1 年以上も前で，もう一方のページの更新時間が最近のものであり二つのページ間で記述の不一致があれば，1 年以上も更新を行われていないページの情報は古いものであり誤っていると推測される．そのため，二つのページ間の更新時間の差を，相当関係を計算する際に考慮する必要がある．二つのページの更新時間の差が小さければ，データ木の内容や構造のわずかな違いでも，対象コンテンツが異なる情報を表している可能性が高い．つまり，相当関係がないと判断する．一方，この二つのページの更新時間の差が大きければ，更新時期の違いによる内容や構造の違いが生じた可能性が高いため，内容や構造の違いがやや大きくても，相当関係の成り立つ可能性がある．

Web ページの更新時間は，Web サーバの応答ヘッダにあるメタデータとして取得できる．しかし，サーバが Last-modified ヘッダを返すように設定されていないページについては，更新時間を取得できないという問題もある．そのような場合には，ページをクロールした時間を更新時間とみなす．2 回以上クロールを行うことで，前回クロールしたページの内容と今回新しくクロールしたページの内容とを比較することで，更新されているかどうかを判断し，更新時間を決定する．また，ページ内のテキストから更新日時の情報を取得するという方法も考えられる．例えば，更新日時や最終更新といった語や日付情報を表す正規表現を手掛かりにして，ページの更新時間を取得できる．このようにして更新時間を取得できれば，更新時間の差を利用できる．

ページ A，B の更新時間を $time(A)$，$time(B)$ と表すことにすると，相当関係を求める際に定める閾値を，その更新時間の差によって定める．更新時間の差が大きいほど，Web コンテンツ間の差異も大きくなる可能性が高いため，閾値が小さくなるようにする．また，閾値は上限と下限を設定する．これは，閾値が 1 に近すぎる場合，相当関係の発見が困難になってしまい，また閾値が 0 に近すぎる場合には相当関係にないものを発見してしまうためである．ページ A，B の更新時間 $time(A)$，$time(B)$ によって相当スコアの閾値 $TimeDifference(A,B)$ を次の式で定義する．

$$TimeDifference(A,B) = \begin{cases} \lambda \cdot \mathrm{e}^{-\frac{|time(A)-time(B)|}{\mu}} & (|time(A)-time(B)| < \mu) \\ \frac{\lambda}{\mathrm{e}} & (\text{それ以外}) \end{cases} \quad (3)$$

ただし，λ と μ はともにパラメータとする．λ は閾値の上限であり，μ は指数関数の逓減率とし，閾値の下限をとるときの更新時間の差とする．下限の値は，上限値の 1/e 倍に設定する．実験では，μ を 500（日）とした．

3.1.4 相当関係の判定

これまでに述べた三つの指標を用いることで，二つのデータ木において相当関係が成り立つがどうかを判定する条件式を定める．(1) 式によって表されるデータ木の類似度と，(2) 式によって表されるリンク関連度に重みをつけ足し合わせたものを，相当スコア $EquivalentScore(t_A,t_B)$ とする．そのスコアがある閾値以上であるときに相当関係が成り立つとみなす．その閾値は，更新時間の差を用いた (3) 式によって求める．相当スコアの計算において，データ木の類似度とリンク関連度では，データ木の類似度の方がより重要であると考えられるため，そ

(注1)：http://www.umechando.com/webex/

ちらの方の重みを大きくする．

ページ A 中のデータ木 t_A，ページ B 中のデータ木 t_B で表される部分において相当関係が成り立つ条件式を，次の式で定義する．

$$EquivalentScore(t_A, t_B) = \alpha \cdot TreeSimilarity(t_A, t_B) + (1-\alpha) \cdot LinkRelevanceDegree(A, B) > TimeDifference(A, B) \quad (4)$$

ただし，α は二つの指標（データ木の類似度，リンク関連度）に対する重みパラメータであり，$0 < \alpha < 1$ とするが，データ木の類似度の重要性から，$\alpha > 0.5$ であることが望ましい．実験では，α を 0.7 とした．

3.2 相当関係部分の探索

Web ページに対して構造化データを階層的に表現したデータ木を構築する．これらのデータ木集合の中から，相当関係にある部分を発見するために，ボトムアップ的にデータ木の部分木同士を比較する．これは，二つのデータ木に相当関係があれば，その部分木にも相当関係が成り立つという性質に基づく．部分木同士の相当スコアを計算し，このスコアが閾値以上であれば，これらの部分木同士に相当関係が成り立つとする．そこで，木の葉ノードから相当関係にある部分を発見し，順次相当関係にある部分が大きくなるように探索を行う．すなわち，再帰的に相当関係にある部分を求め，その最大部分木を発見する．二つのデータ木において相当関係が成り立つかどうかは，(4) 式によって判定する．データ木集合が相当関係にあるとは，集合内の任意のデータ木ペアにおいて相当関係が成立することをいう．例えばデータ木集合 $\{t_A, t_B, t_C\}$ があるとき，t_A と t_B が相当関係ペアであり，かつ t_A と t_C が相当関係ペアであり，さらに t_B と t_C が相当関係ペアにである場合にのみ，t_A と t_B と t_C が相当関係であるとする．

相当関係にあるデータ木について，以下を仮定する．相当関係は再帰的に定義される．

- 最小の相当関係は，葉ノードの相当関係である．
- 高さ 2 以上の相当関係にあるデータ木は，相当関係にある部分木を少なくとも一つはもつ．

これまでに発見した相当関係をもとに，その部分を拡大して相当関係の比較を行う部分を決定する．さらに，比較を行う部分を決定するために，データ木のノード集合の相関を利用する．ここでいうノード集合とは，共通の親をもつノードの集合である．例えば，図 4 では，{教授，助教，秘書}，{教授，准教授，助教}，{教授，助教} などがノード集合となる．同一サイト内，もしくは類似するサイト内において，相関の高い（出現頻度の高い）ノード集合を含む部分木について，相当関係の比較を行う．ノード集合の相関を利用することで，比較を行う部分を効率よく正確に発見できると考える．

以上の議論から，次のような手順で相当関係発見を行う．

Step1. まず，葉ノードにおける相当関係を求める．あるデータ木の葉ノードに対して，他のデータ木のすべての葉ノードに対して一致するかどうかをチェックする．他のデータ木の葉ノードと一致する葉をもたないようなデータ木については，相当関係にある部分が存在しないため，これ以上相当関係部分の探索を行わない．

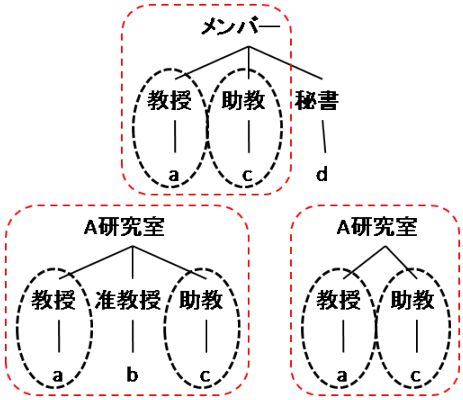

図 4 相当関係の比較を行う部分

Step2. 次に，ある高さ m（m は 1 以上）の部分木における相当関係を求める．このとき，高さ m-1 の部分木における相当関係はすでに求まっている．高さ m-1 の相当関係にある部分木において，その親を根とする高さ m の部分木を考える．

Step3. 高さ m の部分木において，比較する部分を決定する．このとき，高さ m-1 の子部分木は，以下の条件のいずれかを満たす．

- 他の部分木と相当関係にある
- 根ノードが，相関の高いノード集合に含まれる

上の条件を満たさない子部分木については，比較する部分に含めない．

Step4. Step3. によって得られた部分木が，相当関係の候補となる．その中から，任意の部分木ペアについて相当関係が成り立つかどうかを判定する．

Step5. 相当関係が発見されたら，Step2. に戻る．すなわち，発見された相当関係にある部分木を拡大し，さらに相当関係を探索する．相当関係が発見されなければ，探索終了となる．

以上の手順について，例として，図 4 に示した場合を説明する．

- 丸で囲まれた部分木について，相当関係が成り立っているとする．それをもとに，高さ 3 の部分木における相当関係を発見する．相当関係にある部分木の親を根とする木を考える（Step2.）．
- 比較する部分を決定する．“教授”，“助教” ノードを根とする部分木は相当関係にあるため，比較部分に含める．そして，ノード集合の相関を考える．“准教授” ノードについては他のページでも出現する回数は多いが，“秘書” ノードについては，少ない．そこで，比較を行う部分を四角で囲まれた部分に決定する（Step3.）．
- 相当関係の候補にある 3 つの部分木（四角で囲まれた部分）のすべてのペアに対して相当関係が成り立つかどうかを評価し，相当関係を求める（Step4.）．
- 相当関係が成り立てば，さらに探索を続ける（Step5.）．

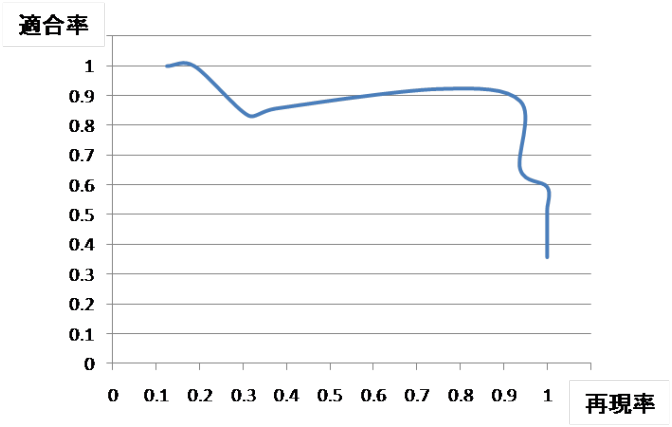

図 5　提案手法の再現率・適合率曲線

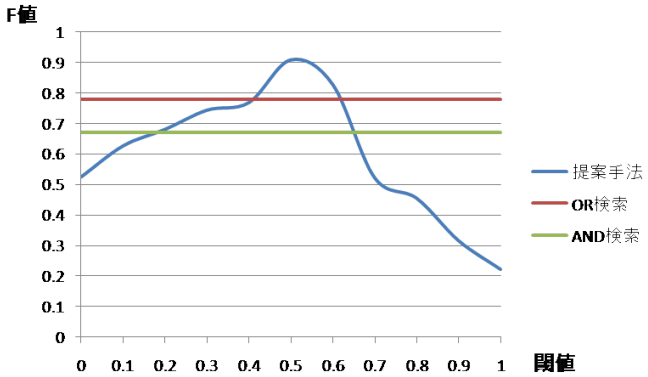

図 6　提案手法とベースラインの F 値比較

表 1　提案手法とベースラインの比較

手法	再現率	適合率	F 値
OR 検索	1	0.64	0.78
AND 検索	0.50	1	0.67
提案手法	0.94	0.88	0.91

4.　評 価 実 験

提案手法によって，実際の Web ページから相当関係を発見することができるかを確認するため，実験を行った．今回の実験では，ある大学の研究室の教員（構成員）情報を対象にして，相当関係を発見した．実際には，複数の Web サイトの全ページを入力とし，すべてのページに対してデータ木を構築し，提案手法を適用するのが望ましいが，今回の実験では簡単のため，ある研究室の構成員情報を事前に与え，構成員名の値を含んでいるような Web ページを指定されたサイトから検索し，対象となるページを絞り込んだ．検索によって得られたページに対し，情報抽出を行い構築されたデータ木を実験に使用した．コンテンツ間で記述の不一致がある場合にも相当関係を発見できるかを検証するため，ページ間で記述の不一致がみられる 10 ページを対象にした．この 10 ページ間の任意の 45 ペアにおいて，相当関係であるかどうかを人手で判断した結果，16 ペアあると集計した．実験の評価尺度としては，再現率，適合率，F 値を用いた．

4.1　実 験 結 果

実験を行う際に決定すべきパラメータは，閾値の上限値である λ と，指数関数の逓減率である μ の二つがあるが，今回の実験では μ を 500 に固定し，λ を 0 から 1 まで 0.1 ずつ変化させることで実験を行った．閾値（の上限）を変化させることによる，再現率・適合率の変化を表すグラフを図 5 に示す．

また，提案手法のベースラインの手法として，単純なキーワードマッチング（OR 検索，AND 検索）によって相当関係を発見し，比較を行った．OR 検索では，ある研究室の教員（構成員）名をクエリとして OR 検索の結果得られたページのすべてのペアに対して相当関係であるみなした．そのため，再現率を 1 に近づけることができるが，相当関係にないページについてもページを取得し相当関係であると誤判断するため，適合率は低くなる．AND 検索では，検索によって得られた 10 ページの中で，教員（構成員）名の組が同じであるページのペアに対して相当関係であるとみなした．そのため，適合率を 1 に近づけることができるが，相当関係にあるページ間で少しでも差異があれば，相当関係でないと判断するため，再現率は低くなる．

提案手法とベースライン手法による F 値の比較結果を図 6 に示す．また，ベースライン手法による再現率，適合率，F 値と，提案手法の F 値で最も高かった結果を表 1 にまとめた．

4.2　考　　察

ベースライン手法と比較すると，提案手法は精度よく相当関係を発見することができた．これは，相当関係を発見する際の指標として，内容だけでなく，（データ木の）構造を考慮したためであると思われる．単純なキーワードマッチングによる OR 検索では，一部分の内容のみに注目し相当関係を発見したため，適合率が低くなった．今回の実験では，相当関係のペアは 45 ペア中 16 ペアと 4 割程度であり，OR 検索による適合率も割と高かったが，実験で対象とするページ数をさらに増やした場合には，相当関係にないページも検索によって多く得られることにより適合率が低下するため，提案手法のさらなる優位性が期待される．また，適合率を上げるために，条件を厳しくして検索を行う AND 検索では，記述の不一致がみられる場合には相当関係を発見できなかった．それに対し，提案手法では，構築したデータ木の内容・構造の差異を許容できるため，コンテンツ間で記述が一致していない場合にも相当関係を発見できた．

今回の実験では，相当関係を発見する際に三つの指標（データ木の類似度，リンク関連度，更新時間）のすべてを利用して評価を行ったが，各指標それぞれについて有効であったかどうかは検証されていない．よって，今後は，各指標（特に，リンク関連度，更新時間）が相当関係の発見に有効であるかをさらに検証しなくてはならない．

5.　応用システム

本研究では，Web コンテンツの相当関係を発見する手法を提案した．Web コンテンツの相当関係を発見することで，次のような二つの応用システムが考えられる．

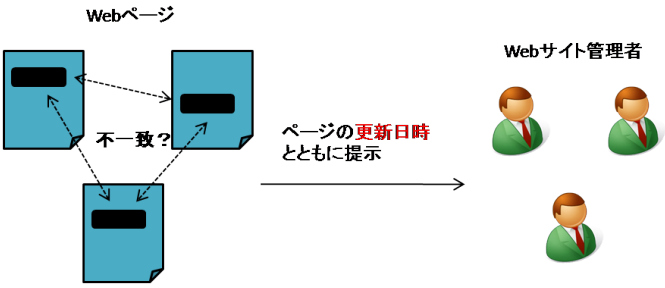

図 7　応用システム 1：不一致箇所の検出

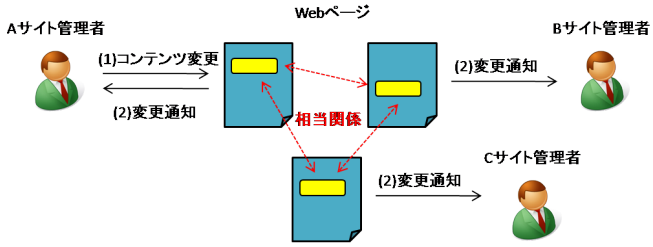

図 8　応用システム 2：コンテンツの変更通知

5.1　不一致箇所の検出

複数の Web ページ群から発見された相当関係にある Web コンテンツの組に対して，データ木の構造・内容を比較することで，記述の不一致箇所を推定して提示するといったシステムが考えられる（図 7）．一致していない記述箇所とともに，相当関係にある各 Web ページのコンテンツ・更新時間を提示することを行い，各サイトの管理者は提示された情報をもとに，コンテンツの記述に誤りがあるかどうかを実際に判断し，もしそうであれば Web ページに対して修正を加えるなどとすればよい．

相当関係にある Web コンテンツの組を発見する際には，リンク構造および更新時間も考慮したが，相当関係にあるコンテンツの組の中で記述の不一致があるかどうかはデータ木の類似度のみから判断する．データ木の類似度がある閾値以下であるときに，二つのデータ木において記述の不一致が生じているとみなす．相当関係であるかどうかを判定する閾値については小さく設定することで，網羅的に相当関係を発見でき，記述の不一致が生じているかどうかを判定する閾値については大きく設定することで，記述の不一致があまりなくても検出できる．

5.2　コンテンツの変更通知

他の応用として，既存研究 [2] [3] で提案されているような Web サイト管理を行うシステムが考えられる（図 8）．Web コンテンツの相当関係にある組を求めることできれば，その Web コンテンツの組において相当関係が成立しているという制約を設けることにする．この制約に基づいて，あるコンテンツに対して変更が加えられた場合に，そのコンテンツと相当関係にあるコンテンツに対しても変更を加える必要があるかもしれないということを各サイトの管理者に通知することができる．

6.　おわりに

本稿では，複数の Web ページ間で同じような記述をした部分の特定，すなわち相当関係にあるようなコンテンツの組を発見する手法を提案した．Web ページに対して構造化データを階層的に表現したデータ木を構築し，その中から相当関係が成り立つ部分を発見する．リンク構造および更新時間を考慮することで，データ木の構造・内容に違いがある場合にも相当関係を発見できる．大学の Web サイトを対象に，相当関係を発見する実験を行い，本手法の有効性を検証した．相当関係にある Web コンテンツの組を発見することで，異なるページ間における記述の不一致の検出が可能になる．

最後に，今後の課題として，以下のようなものが挙げられる．

(1) 提案手法の汎用性の向上

本研究では，データ木を構築するために，スキーマを指定することによりデータの抽出を行った．また，対象にした教員情報のページは，比較的同じような構造をもつページが多かった．そのため，データ木を比較する際には，スキーマのマッチングを行う必要がなかった．しかし，現実的に大規模なページから抽出を行う場合には，人手によってスキーマを指定することが困難になり，教師なしで抽出を行うと，多様なスキーマを抽出するためスキーマのマッチングが必要となる．提案手法の汎用性を向上させるために，スキーマのマッチングを考慮したデータ木の比較方法を考えなければならない．

(2) 追加実験

今回の実験で対象にした Web ページ数は少ないものであったため，十分に手法の有効性を検証できていない．今後はより大規模な実験を行い，多様な Web ページに対しても手法が適用可能であるかどうかを検証する必要がある．また，大学の Web サイト以外にも，異なるページ間で Web コンテンツの相当関係が成り立つ例として，企業の Web サイトにある製品情報などがあるが，このような Web サイトに対しても，本手法による相当関係の発見が有効であるかどうかを検証してきたい．

(3) 表記ゆれへの対応

データ木の類似度を計算する際，ノード同士の比較は，その値が完全に一致するときにのみ，ノードが一致するものとみなした．しかし，それではノード間の軽微な誤りや，表記のゆれといったものに対しては，不一致とみなす．例えば，氏名において，漢字の旧字体が用いられている場合や，所属となる組織名において省略がされている場合に対して問題が起きる．そこで，同意語辞書を利用したり，単語の共通文字列や文字列間の編集距離を考えるなどして表記のゆれにも対応しなければならない．また，記述量が多い場合には，単純に単語が一致するかどうかではうまくいかないため，コサイン類似度を用いるなどの方法が考えられる．

(4) 応用システムの実現

第 5 章で述べたように，提案手法を実際の Web ページに適用し，相当関係にあるコンテンツの組を発見することで，二つの応用システムが考えられる．発見されたコンテンツの相当関係に基づき，(1) 記述の一致していない部分を検出するシステム，(2) コンテンツの変更を通知するシステムを実際に構築し，その有用性を確認したい．

(5) 提案アルゴリズムの改良

Web サイトが大規模になるほど，構築されるデータ木も大き

くなり，また入力となる Web ページが多いほど，比較を行うデータ木が増える．すると，相当関係部分の発見を行うアルゴリズムの計算量が大きく関わってくるため，効率よく発見する手法についても検討していきたい．

謝　　辞

本研究の一部は，科研費 (No.20300042，No.20300036) の助成を受けたものである．

文　　献

[1] Ion Muslea, Steve Minton, Craig Knoblock, “A hierarchical approach to wrapper induction”, International Conference on Autonomous Agents, pp. 190-197, 1999.

[2] 澤菜津美, 森嶋厚行, 飯田敏成, 杉本重雄, 北川博之, “コンテンツ一貫性制約を用いた Web サイト管理手法の提案”, 電子情報通信学会第 18 回データ工学ワークショップ (DEWS2007) 論文集, 2007

[3] 高橋公海, 森嶋厚行, 松本亜季子, 杉本重雄, 北川博之, “Web コンテンツ管理のための一貫性制約発見手法”, DBSJ Journal, Vol. 7, No. 3, pp.25-30, 2008.

[4] Bing Liu,「Web Data Mining, Exploring Hyperlinks, Contents, and Usage Data」, Springer-Verlag, chapter 9, pp. 323-380, 2011.

[5] Yanhong Zhai, Bing Liu, “Web data extraction based on partial tree alignment”, Proceedings of 13th International World Wide Web Conference (WWW2005), pp. 76-85, 2005.

[6] Nitin Jindal, Bing Liu, “A Generalized Tree Matching Algorithm Considering Nested Lists for Web Data Extraction”, Proceedings of the 10th SIAM International Conference on Data Mining, pp. 930-941, 2010.

[7] 寺島慎太郎, 天笠俊之, 北川博之, “XML データにおける類似極大部分木の効率的な探索”, 第 3 回データ工学と情報マネジメントに関するフォーラム (DEIM2011) 論文集, 2011.

[8] Philip Bille, “A Survey on Tree Edit Distance and Related Problems”, Theoretical Computer Science, Vol. 337, pp. 217-239, 2005.