Webコンテンツ一貫性管理支援ツールの開発

高橋 公海† 澤 菜津美‡ 森嶋 厚行‡ 杉本 重雄‡ 北川 博之††

筑波大学 図書館情報専門学群† 筑波大学大学院 図書館情報メディア研究科‡ 筑波大学大学院 システム情報工学研究科††

1. はじめに

近年，Webサイトを通じた情報発信が広く普及し，コンテンツの量も増加している．Webの特徴の一つは分散管理であるが，一方で，その特徴がコンテンツ一貫性の維持を困難とする一因となっている．例えば，大学の研究室のWebサイトでは，各構成員が自分のホームページ上で研究論文リストを公開することが多いが，これらの論文リストの間には矛盾が多く見られる等といった問題がある．

一般に，コンテンツの一貫性を管理するためには，バックエンドにDBシステムを配置し，DBに格納されているデータからWebページを作成するアプローチがとられる．しかし，筑波大学のWebサイトを対象とした我々の予備調査[1]では，バックエンドDB等をもたずに手作業で管理されているWebサイトも，数多く存在する．

これまで我々は，DB等をバックエンドに持たないWebコンテンツの一貫性管理の支援を目的として，各種要素技術の研究開発を行ってきており，現在は，それらの要素技術を統合して利用するための支援ツールを開発中である．本稿では，開発中のWebコンテンツ一貫性管理支援ツールの概要を説明する．また，既存のWebコンテンツ間に存在する包含従属性の発見を支援するための手法について提案する．

2. 明示的な一貫性制約を用いたWebコンテンツ管理手法

図1は明示的な一貫性制約を用いたWebコンテンツ管理の仕組みを表したものである．以下にその手順を述べる．(1) まず，利用者がコンテンツ一貫性制約を登録する．(2) 制約が登録されると，システムは定期的もしくはWebサイトの更新が行われた際などにWebサイトのチェックを行い，先に発見しておいた制約と照らし合わせて，制約が破られていないかどうか調べる．(3) その際，もし制約違反を発見したら，Webサイト管理者に報告もしくは自動修正を行う．

現在，一貫性制約としては，包含従属性[2]を対象としている．また，Webコンテンツのラッピングを行うための軽量ラッピング言語Wraplet[3]を提案している．

3. プロトタイプシステム

本研究で作成した一貫性管理支援ツールのプロトタイプの画面を図2に示す．本システムはJavaで実装した．本システムは現在下記の二つの機能を持つ．(1) 入力として

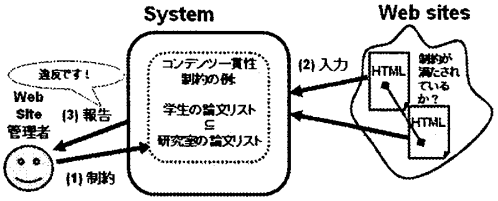

図1 コンテンツ一貫性制約を用いたWebサイト管理

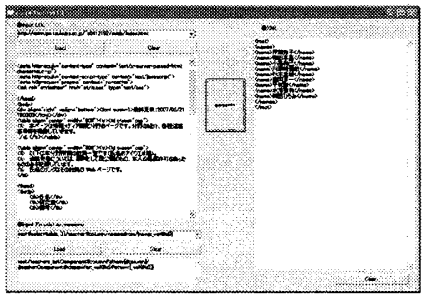

図2 一貫性管理支援ツールプロトタイプ画面

HTMLページのURLとWraplet式を受け取り，ラッピング結果のXMLデータを出力する．(2) 指定されたWebサイトに含まれるWebページ間の包含従属性の発見支援を行う．これについては次章で説明する．(3) Webページのコンテンツ間に，指定された包含従属性が成立しているかどうかの判定を行う．

4. Webコンテンツ間の包含従属性発見支援

本ツールを利用するためには，Webコンテンツに関する包含従属性をツールに与える必要がある．しかし，Webコンテンツを構成するページ数が増えると，手作業で一つずつ発見することは現実的ではない．本節では，Webコンテンツに関する包含従属性の発見を支援する手法を提案する．

全体像. 本ツールで言うWebコンテンツに関する包含従属性とは，Webページを構成するHTMLもしくはXML要素を単位とし，2つの異なる要素に含まれるコンテンツ間に包含関係が存在しなければならない事を表す．例えば，研究室の論文リストを表す要素Aのコンテンツと，その研究室の構成員の論文リストを表す要素Bのコンテンツ間に包含従属性が存在するといったことである．より具体的には，$C(x)$ を要素 x のコンテンツを表す集合としたとき，$C(B) \subseteq C(A)$ が必ず成立することを表す．上の例では，$C(B)$ と $C(A)$ は，それぞれ論文の集合である．

本手法では，Webサイトに含まれる全てのWebページに含まれる全ての要素の組合せに対して，包含従属性の証拠となる包含関係が存在するかどうかの判定を行う．既に，集合 X と Y が与えられたとき，$X \subseteq Y$ が成立するかどうかを判定する効率よい手法は存在する[4]．しかし，我々の問

Development of a Support Tool for the Maintenance of Web Content Integrity

Masami Takahashi† Natsumi Sawa‡ Atsuyuki Morishima‡ Shigeo Sugimoto‡ Hiroyuki Kitagawa††

School of Library and Information Science, Univ. of Tsukuba.†

Grad. Sch. of Library, Information and Media Studies, Univ. of Tsukuba.‡

Grad. Sch. of Systems and Information Engineering, Univ. of Tsukuba.††

```
1. Input: refValues, depValues, p
2. Output: Is depValues ⊆ refValues satisfied?
3.
4. C = p * |depValue|; // 許容する例外数
5. while  depValues has next value  do
6.    currentDep := depValues.next() ;
7.    if refValues is empty then    return false;
8.      while true do
9.          currentRef := refValues.next() ;
10.         if currentDep = currentRef then    break;
11.         else if currentDep < currentRef then
12.           C := C - 1;
13.           if C == 0 true then    return false;
14.           if depValues has next then
15.              currentDep := depValues.next();
16.              currentRef := refValues.prev();
17.           else    break;
18.       else if refValues has no next value then    return false;
19. return true;
```

図 3 アルゴリズム

題設定では，ページ x が与えられたときそのページのコンテンツが表す集合 $C(x)$ を求めることは一般には自明ではない．また，Web ページのコンテンツには間違いも多く，厳密な包含関係の判定ではうまくいかない．したがって，本手法では，要素の内容を文字 N-gram などで単純に分割し，かつ，一定のエラーを許すことにより，包含関係が存在する可能性が高い組合せを求める．

アルゴリズム．図 3 は，二つの集合 $depValues$ と $refValues$ が与えられたとき，$depValues \subseteq refValues$ が成立するかどうか判定するアルゴリズムである．これは論文[4] でのアルゴリズムを拡張したものである．本手法では，二つの要素 dep と ref に関する包含関係を調べるとき，$depValues$ を $C(dep)$ の近似として dep のコンテンツの (ソートされた)N-gram 集合とし，$refValues$ を $C(ref)$ の近似として ref のコンテンツの (ソートされた)N-gram 集合とする．また，入力として，$depValues$ と $refValues$ の他に許容度の割合を表す 0 以上 1 以下の値 p を取る．

まず 4 行目で，$depValues$ のうち何個が $refValues$ に含まれていなくても許容するかのカウンタ C を計算する．それ以降では，$depValues$ に含まれる各値が $refValues$ に含まれているかどうかを順に判定していき，含まれていない値を見つけるたびに C を 1 つ減じる．そして，C が 0 になれば，包含関係は存在しないとして false を返す．

値が含まれているかどうかの判定は，入力がソート済みであるという性質を利用して，効率よく行う．まず，両方のデータ集合の最も小さい項目から順に参照しながら比較していく．もし $depValue$ 中の現在の値が $refValue$ 中の現在の値と等しい (10 行目) ならば，同じものが含まれているため，$depValues$ と $refValues$ を次の値に進める．もし $depValue$ 中の現在の値が $refValue$ 中の現在の値よりも小さい (11 行目) ならば，その値は $refValue$ 中に含まれていないと判定できる．このようにして，$depValues$ の値が $refValues$ に含まれるかの判定を行う．

5. 実　　験

本手法の適用実験を行った．具体的には，実際の Web サイトから作成した次の二つのデータセットに対し，本手法を適用した．(1) 既に包含関係があることが分かっている要素のペアから 10 組を選択．(2) 包含関係が無いことが分かっている要素のペアから 10 組を選択．

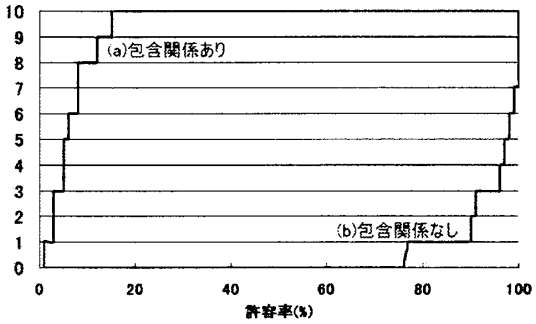

図 4 実験結果

結果を図 4 に示す．本図は，許容率 p を 0 から 1(100%) まで 0.01(1%) 刻みで実験を行い，包含関係があると初めて判定された値をカウントした累積度数分布である．

既に包含関係があると分かっている要素間では許容率 15%以下で全て包含関係があると判定された．また，包含関係がないと分かっている要素間では，許容率が 76.1%になるまで包含関係があるとは判定されなかった．このことから，許容率を適切に判定することで，包含関係が成立する (すなわち包含従属性が存在する可能性が高い) 組合せを求めることが可能と推測できる．

6. まとめと今後の課題

本稿では，バックエンドに DB を持たないような Web サイトであってもコンテンツの一貫性管理を可能にするためのツールの開発について説明した．また，特に，Web コンテンツ間の包含従属性の発見を支援するためのアルゴリズムについて提案した．今後の課題としては，本ツールの機能強化や，本ツールおよび提案アルゴリズムを用いた実際のコンテンツ一貫性管理の有効性の評価などがあげられる．

謝　　辞

本研究の一部は科学研究費補助金特定領域研究 (#19024006)，科学研究費補助金基盤研究 (B)(#19300081) による．

参 考 文 献

1) 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)，7 pages，2007 年 3 月．
2) Serge Abiteboul, Richard Hull, Victor Vianu: Foundations of Databases. Addison-Wesley 1995.
3) Natsumi Sawa, Atsuyuki Morishima, Shigeo Sugimoto, Hiroyuki Kitagawa. Wraplet: Wrapping Your Web Contents with a Lightweight Language. Proc. of IEEE The Third International Conference on Signal-Image Technology and Internet-based Systems (SITIS' 2007).
4) J. Bauckmann, U. Leser, F. Naumann. Efficiently Computing Inclusion Dependencies for Schema Discovery. Proceedings of the 22nd International Conference on Data Engineering Workshops (ICDEW'06), 3-7 April 2006.