2004－NL－162　(7)
2004／7／15

メタデータ付与のための住所録自動生成

村山 紀文[†]　南野 朋之[†]　奥村 学[‡]

概　要

位置情報は Web 上の情報の組織化において有用な情報と考えられるが, 現状では, 陽に位置情報が明記されていないため, 位置情報を利用できない場合が多い. しかし, 店名などの固有名はそれ自体は位置情報を直接的に示すものではないが, 位置情報と関連した固有名と考えることができる. これらの固有名に対し, 位置情報をメタデータとして付与することが出来れば, その位置情報を利用することが出来る. 本研究では, そのようなメタデータ付与のために必要な, 固有名と対応する位置情報のデータベースを, Web 上の文書から自動的に作成する手法を示す. 本稿では特に, その最初のステップとして, 固有名・住所・電話番号の三つ組を Web 上の文書から自動的に抽出する手法を示す.

Automatic Generation of an Address Book for Metadata Annotation

Norifumi MURAYAMA[†]　Tomoyuki NANNO[†]　Manabu OKUMURA[‡]

Abstract

Location information in web pages is very useful to organize information in them. However, many web pages do not include apparent location information in themselves. Though named entities, such as names of stores, are not location information, we think those named entities are related to location information. If we can annotate location information to such named entities, we can use these pages with location information. Therefore, our goal is to extract meta data base for such named entities, that include their location information, telephone number and other information. In this paper, we present a method to automatically constract triples (named entity, its telephone number, its address) from the WWW as a first step of meta data base construction.

1　はじめに

現在 Web 上には多くの情報が存在している. それらの情報をユーザが利用し易くするための方法の 1 つとして, 何らかの軸に基づく情報の組織化がある. たとえば, トピックごとに情報を分類する, 情報を時間軸上に整列する, 含まれている位置情報を元に情報を整理するなどである.

位置情報はこのように, Web 上の情報の組織化において有用な情報と考えられる. 位置情報を利用することで，ユーザは自分の求める地域に限定した情報を検索できたり，見ているページ中に書かれている情報と場所的に近い情報が書かれたページへ移動できたりするようになる.

また, 特に近年では, Web 上に詳細な電子地図が整備されてきており, 位置情報を地図上にマッピングすることが可能になり，また GPS 技術の開発・携帯端末の発達によって，携帯端末によるナビゲーションサービスが数多く展開され始めるなど，位置情報の有用性は大きく高まってきている.

[†]東京工業大学大学院 総合理工学研究科
Interdisciplinary Graduate School of Science and Engineering, Tokyo Institute of Technology
{murayama,nanno}@lr.pi.titech.ac.jp
[‡]東京工業大学 精密工学研究所
Precision and Intelligence Laboratory, Tokyo Institute of Technology
oku@pi.titech.ac.jp

そのため, Web 上の文書に対して, メタデータとして位置情報を付与する研究も活発化してきている [1]. Web 上の文書中に位置情報が明記されている場合, 情報抽出技術の 1 つである固有名 (住所情報) 抽出技術 [2] を用いることで, 文書中の位置情報を抽出すれば, 位置情報をメタデータとして付与することができる.

では, 陽に文書中に位置情報が明記されていない場合, 位置情報をメタデータとして付与することはまったくできないだろうか. たとえば, お店の名前, 病院の名前等の固有名は, それ自体は, 位置情報を明記していないが, 位置情報と関連した固有名と考えることができる. これらの固有名に対し, 位置情報との対応を表すデータベースを作成し, 固有名の出現を元に, 文書に対し位置情報をメタデータとして付与することができれば，より多くの文書に対し位置情報のメタデータを付与することができ, 有用性は大きく増すと考えられる.

そこで, 本研究では, このようなメタデータ付与のために必要な，固有名と対応する位置情報のデータベースを, Web 上の文書から自動的に作成する手法を示す. 本稿では特に，位置情報との対応を含む固有名データベース自動作成の最初のステップとして，固有名・住所・電話番号の三つ組を Web 上の文書から自動的に抽出する手法を示す. 将来的には, 位置情報との対応を含む固有名データベース中には, 付加情報として, 固有名のカテゴリ（飲食店・病院など）・営業時間なども追加していきたいと考えている. このデータベースは Semantic

Web[3] の考えに基づくなら, 文書中に含まれる固有名に対するメタデータベースということになる.

2 関連研究

Web 上の位置情報への関心は, 近年著しく高まってきている. 携帯電話からの GPS 情報付き画像を収集し地図上にマッピングし表示するアプリケーション [4] や, 携帯端末などを用いた地域密着型の情報基盤の開発 [5] などは, 位置情報への関心の高さを示している. 本研究の目的である固有名への位置情報データ付与は, 上記のような情報処理技術において利用可能な情報源を拡大することにつながる.

本研究で目標とするような住所録型のデータベースは，Yahoo!電話帳[1]やインターネットタウンページ[2]など, 既にいくつか作成されており，Web 上で公開されているものも多い．また，ぐるなび[3]やグルメぴあ[4]のようなポータルサイトでは営業時間や駐車場の有無などの, より多くの付加情報を含めたデータを公開している.

しかし，これらのデータは人手で構築されたものであり，作成に多大なコストを要する．また、Yahoo!電話帳のような住所録では固有名の表記の揺れに対応できなかったり，電話番号・住所以外の付加情報が少ないという問題点がある．ポータルサイトでは上記の問題は少ないが，データベースが飲食店に対象が限られていたり規模が小さいなどの問題がある.

本研究では Web から情報抽出することで，表記の揺れやさらなる付加情報を取得することが出来る．また, それ以外にも網羅性の高さや新規の情報を素早く獲得出来る点などで，既存のデータベースよりも有用なデータベースになることが期待される.

本研究の関連研究としては，[6] があげられる．[6] では，ユーザから与えられた固有名を元に Web 上の文書を検索し，検索結果から固有名に対して正しいと思われる住所・電話番号・郵便番号を提示するシステムを構築している．この手法では固有名が与えられていることが前提であるので固有名抽出を行う必要がないが, 本研究の手法では固有名も Web 上から獲得するために文書中から固有名部分を特定しなければならない．また, この手法では抽出された住所・電話番号がある１つの固有名に対応しているかどうかだけを考えればよいが, 本研究では複数の固有名・住所・電話番号に関して組み合わせを考えなければならない.

3 手法概要

本研究は Web ページから固有名・住所・電話番号の三つ組を自動抽出することを目的としている．このような三つ組が書かれているページは大きく次の二つに分類することが出来る．一つの三つ組だけが書かれているページと, 複数の三つ組が併記されているページである. 前者には店舗の HP やポータルサイトの中の一店舗の紹介ページが該当し, 後者にはあるカテゴリに属する店舗の情報を集めたページや, 企業の支店一覧などが該当する．本論文では特に後者のような複数の三つ組が併記されているページのうち, 三つ組が構造的に規則正しく書かれているようなページを “一覧” 型と呼ぶことにし, これらのみから三つ組を抽出することにする.

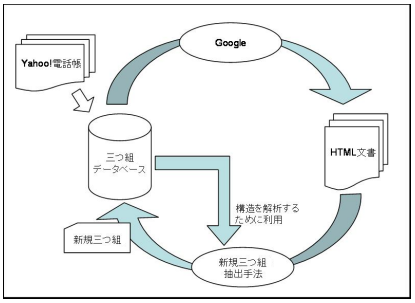

図 1: 抽出手法概要

Web ページから固有名・住所・電話番号の三つ組を抽出するとき，住所・電話番号は地名データベースとのマッチングやパターンマッチングを行うことで抽出できるが，固有名の抽出はそのようなマッチング規則を簡単に作ることが出来ないため, 非常に難しい問題である．また，１つのページに三つ組が複数併記されている場合は，正しい組み合わせを考えることも問題となる．同時に，抽出対象となる三つ組が記載された Web ページを効率良く検索する方法も問題となる.

そこで本研究では，あらかじめ三つ組が登録されたデータベースを用意しておき, 上で述べた “一覧” 型ページにおいて, 既にデータベースに登録されている三つ組がどのように記述されているかを解析し，その構造を利用することで新たな三つ組を獲得し, 三つ組のデータベースを拡張していくことを考える．このような方法を採ることによって, 前述したような固有名抽出問題や組み合わせの問題を格段に簡単にすることが出来る. また, データベース中の三つ組からクエリを作成し, 検索エンジンで検索することにより, 三つ組が書かれているページを効率よく獲得することが出来る.

図 1 に抽出手法のフローチャートを示す．まず, 既存のデータベース中の三つ組を元に検索を行い, 抽出対象となる HTML 文書を収集する．そこで得られた文書のうち “一覧” 型の文書を対象に, 既存の三つ組を利用して構造解析を行う．その解析結果を用いて, 新規の三つ組を抽出する．本手法は抽出された三つ組を既知の三つ組に加え，手法を再帰的に適用することで，新たな抽出対象ページを獲得していくことが出来，また既に獲得済みのページからも新たな構造を解析することが出来るようになる．この繰り返しによって，三つ組のデータベースを拡大していく.

3.1 使用データ

本研究では，種となる三つ組のデータベースとして Yahoo!電話帳のデータを利用する[5]．更に後の処理のために，日本郵政公社が公開[6] している地名データを利用

[1] http://phonebook.yahoo.co.jp/
[2] http://itp.ne.jp/servlet/jp.ne.itp.sear.SCMSVTop
[3] http://www.gnavi.co.jp
[4] http://g.pia.co.jp/

[5] 実験では，一部のみ使用
[6] 郵便局ホームページ `http://www.yuubinkyoku.com/`

名称	電話番号	住所
東工亭	045-xxx-xxxx	神奈川県横浜市緑区x丁目x-x
東工軒	045-xxx-xxxx	神奈川県横浜市緑区xx-x
東工本舗	045-xxx-xxxx	神奈川県横浜市緑区xx-x

図 2: 抽出対象ページ例 1

名称	東工亭	電話番号	045-xxx-xxxx
住所	神奈川県横浜市緑区x丁目x-x		
名称	東工軒	電話番号	045-xxx-xxxx
住所	神奈川県横浜市緑区x丁目x-x		
名称	東工本舗	電話番号	045-xxx-xxxx
住所	神奈川県横浜市緑区x丁目x-x		

図 3: 抽出対象ページ例 2

し，各三つ組の住所を「都道府県」「市区町村」「地域名」とそれ以降に続く「番地など」に分割し保存した．

4 抽出対象ページ検索

まず抽出対象とするページの獲得を行うが，これには Google Web APIs [7] を利用した．

我々の手法では，新規の三つ組を抽出するためには，後述するように，抽出対象となるページに既知の三つ組が少なくとも 2 つ以上含まれている必要がある．しかし，既知の三つ組の全ての組み合わせを検索しようとすると，検索を相当な回数繰り返さなければならず，効率的とは言えない．

本手法では，まず既知の三つ組を少なくとも 1 つ含んでいるページを検索し，情報抽出を行う際に改めて既知の三つ組を複数含んでいるかどうかをチェックするという方針をとる．

実際には検索クエリは既知の三つ組から以下のような形式で生成した．

「“電話番号” 市区町村名」

電話番号が “” で囲まれているのは，フレーズ検索であることを示している．

また，本手法は現在の所 HTML 形式の文書のみを対象としているので，HTML 形式でないページは検索対象外とする．また，全てが既知であると判明している Yahoo!電話帳内のページも対象外とする．

5 抽出手法

獲得された HTML ファイルにおいて，既知の三つ組が複数同じような構造で記述されている場合，未知の三つ組もまた同じような構造で書かれているものと考えられる．

例として，図 2 のようなページを処理することを考えてみる．ここで,「東工亭」と「東工軒」の三つ組がそれぞれ既知であったとすると，それらの構造からテーブル構造の 1 列目が固有名，2 列目が電話番号，3 列目が住所であり，同じ行にあるものが 1 つの組として考えられるということがわかる．ここから，未知である「東工本舗」の三つ組を獲得することが可能になる．しかし，図 3 の場合は同じ 2 列目でも奇数行が固有名，

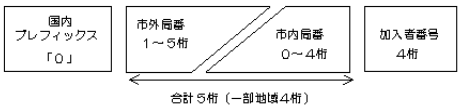

図 4: 固定電話の電話番号

偶数行が住所となっており，例 1 のように構造を掴むことが難しい．

本研究では，例 2 のような複雑な構造のファイルも同様に処理するために，電話番号を手がかりとし，DOM Tree に基づいたパス表現を用いた抽出手法を提案する．本研究の手法はリスト・テーブルなどの特定の構造に依存しない手法であり，様々な構造に対して適用することが可能である．

抽出手法は，主に次のステップにわけることが出来る．

1. 前処理
2. 電話番号のパターンマッチング
3. 既知固有名・住所のマッチング
4. 新規電話番号の特定
5. 新規の固有名・住所の特定
6. 新規三つ組抽出

以下でそれぞれのステップの詳細を述べる．

5.1 前処理

取得した Web ページに対して HTML Tidy[8] を適用し，HTML 文書を well-formed な XML 文書にする．これにより，開始タグと終了タグの対応が取れていることが保証され，以降の処理を適切に進めることが出来る．

また，同時に電話番号に使われやすい全角数字や全角ハイフン・全角括弧はパターンマッチングを行いやすくするために全て半角に直した．

5.2 電話番号のマッチング

本手法では，まず始めに簡単にマッチングを行いやすい電話番号部分を特定し，タグ付けを行う．既知の三つ組に含まれる電話番号が見つかった場合は，対応する固有名と住所を特定していくという方法をとる．

電話番号をマッチングさせるための正規表現は，図 4[7]を参考にして作成した．

単純なルールのため，例えば “010-000-0000” のようなシリアル番号などの（電話番号ではない）数列があった場合にもマッチしてしまう．だが，このような数列は後のステップでのチェックにより取り除かれることが期待される．

正規表現にマッチした電話番号のうち，既にデータベースに登録されている既知のものに関しては ⟨tel id=“電話番号”⟩ タグで囲み，未知の電話番号は ⟨tel⟩ タグで囲む．

[7]総 務 省　HP(http://www.soumu.go.jp/joho_tsusin/top/tel_number/fixed.html) 内より抜粋

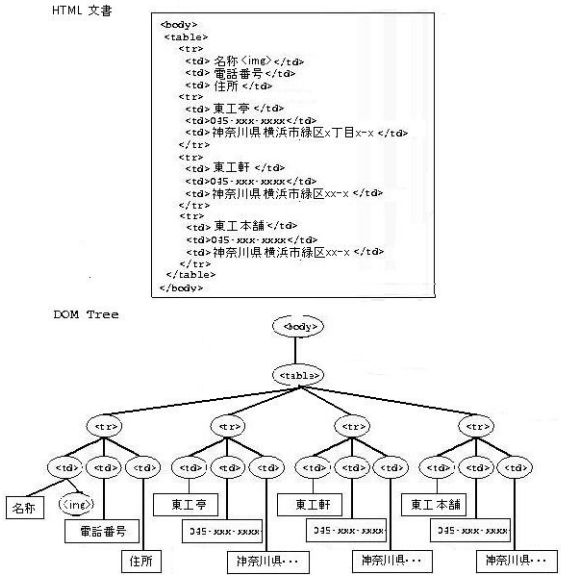

図 5: HTML ファイルと DOM Tree

5.3 既知固有名・住所のマッチング

次に，マッチした電話番号のうち既知のものに関して，対応する固有名と住所が含まれているかをチェックする．

固有名に関しては，データベース中に登録されている固有名との完全マッチでそのチェックを行う．住所に関しては,「都道府県名」「市区町村名」は省略される場合もあるため,「地域名」と「番地など」が一致していればマッチしているものとする[8]．

上記のようにしてマッチングを行い，正しく対応のとれた既知の固有名・電話番号・住所が見つかった場合にのみ，固有名・住所をそれぞれ ⟨name id=“電話番号”⟩⟨address id=“電話番号” ⟩ タグで囲む．複数マッチした場合は，HTML ファイル中で電話番号から一番近いものを対応するものとする．

同一ページ内から既知の三つ組が 2 つ以上見つかった場合のみ，以降のステップを適用する．

5.4 新規電話番号の特定

DOM 構造とパス

既知三つ組と未知の電話番号候補に対してタグを振った時点で，HTML 構造の解析を行う．HTML 文書の構造は DOM(Document Object Model) 構造 [9] で表現することが出来る．DOM 構造は DOM Tree という木構造で示される．例として, 図 2 に対応した HTML 文書と DOM Tree を示す（図 5）.

本研究では，HTML 文書から DOM Tree を構築したあと，DOM Tree における各ノードを root（⟨body⟩ タグ）からのパスによって表現し，解析・抽出に用いる．例えば図 5 の DOM Tree 表現において斜線の入った部分を表すパスは

/body/table/tr/td/

となる．

以後は, 図 5 を例に挙げながら説明を行っていく．

パスによる特定手法

パス表現を用いて，新規電話番号を特定する．

まず，既知の電話番号（⟨tell id=“電話番号”⟩ のタグが振られている）へのパスを求める．図 5 の例では，既知の「東工亭」と「東工軒」の電話番号へのパスである

/body/table/tr/td/

/body/table/tr/td/

の 2 つを獲得することが出来る．

このように，複数同じパスが得られた場合，そのパスは現在処理中の HTML 文書において電話番号が記述される場所へのパスであると考えることが出来る．

この複数の既知電話番号に共通なパスを辿っていった場所に未知電話番号（⟨tell⟩ のタグが振られている）が存在していれば，その電話番号は三つ組として獲得可能な電話番号であると考える．

図 5 の例では共通パス/body/table/tr/td/を辿っていくと未知電話番号である「東工本舗」の電話番号を得ることが出来る．

5.5 新規の固有名・住所の特定

パスの拡張

固有名と住所の特定も電話番号と同じようにパスを用いて行うが，電話番号に用いたようなタグ名だけのパス表現では不十分である．

図 5 の例では前述のパス表現を用いると固有名・住所共に

/body/table/tr/td/

という表現になってしまう．このような大雑把なパス表現では, 固有名・住所が記述されている場所を正確に特定するのは難しい．

そこで，より詳細なパスを示すため次のようなパスを定義する．

/ タグ名 [値 1 ：値 2] /

以後，これを完全パスと呼ぶ．ここで，値 1 は「同じ階層で何番目の子か」の値であり，値 2 は「同じ階層の同じタグの中で何番目か」の値である．例えば図 5 の DOM Tree 表現において「東工亭」が記述された部分を表すパスは

/body[0,0]/table[0,0]/tr[1,1]/td[0,0]/

となる．

完全パスによる特定手法

新たに定義した完全パスを用いて, 固有名部分と住所部分の特定を行う．

まず，前のステップで未知の電話番号を導出するのに使われた既知電話番号に関して，それぞれに対応する固有名・住所への完全パスを得る．そして得られたそれぞれの完全パスについて，電話番号からの相対パスを計算する．図 5 の例で「東工亭」,「東工本舗」に関して考えると，以下のように相対パスを計算することが出来る．

[8]実際は, Web 上に出現する固有名と住所には表記の揺れが存在する．今後の課題として，この表記の揺れに対応した，より柔軟なマッチング方法を考える必要がある

「東工亭」
電話番号　/body[0,0]/table[0,0]/tr[1,1]/td[1,1]/
固有名　/body[0,0]/table[0,0]/tr[1,1]/td[0,0]/
↓
相対パス　../td[*,-1]

「東工本舗」
電話番号　/body[0,0]/table[0,0]/tr[3,3]/td[1,1]/
固有名　/body[0,0]/table[0,0]/tr[3,3]/td[0,0]/
↓
相対パス　../td[*,-1]

この場合の td タグのように同じ階層が同じタグで構成されていた場合，相対的な構造的関係はそのタグのみに依存すると考えることが出来る．よって，上記の場合は相対パスは値２だけで構成することが出来る．

逆に，次の場合のように同じ階層が同じタグで構成されていなかった場合は,「同じタグの中で何番目か」という情報は重要ではない．よって，相対パスは値１だけで構成される．

電話番号　/body[0,0]/p[1,0]/ul[2,0]/li[1,1]/
固有名　/body[0,0]/p[1,0]/h3[0,0]/
↓
相対パス　../../h3[-2,*]

上記のようにして構成された相対パスのうち，同じページ内から複数の同一相対パスが見つかった場合，それは新規の三つ組にも成立するパスであると考え，未知の電話番号に対して適用する．図 5 の例では，未知の「東工本舗」の電話番号から

電話番号　/body[0,0]/table[0,0]/tr[2,2]/td[1,1]/
相対パス　../td[*,-1]
↓
固有名　/body[0,0]/table[0,0]/tr[2,2]/td[*,0]/

のようにして，正しく対応する固有名「東工本舗」へのパスを得ることが出来る．住所に関しても，同様に正しくパスを得ることが出来る．

前後文字による固有名特定

このようにパスを得ることが出来たとしても，パスによって示される固有名や住所には，余分な文字列がついてしまっていることがある．住所に関しては後処理で地名データベースと照合することによって余分な部分を排除することが出来るが，固有名に関しては後処理で排除することは難しい．固有名に余分な文字列がついている例としては，次のようなものが挙げられる．

- ○○亭 9:00〜18:00
- 和食 ○○亭 駐車場あり
- 和食「○○亭」
- 「○○亭（平仮名表記）」
- …· 今日は「○○亭」というところに …·

しかし，固有名は同一ページ内では「」で囲まれていたり，空白で区切られているなど同じ特徴で表記されやすいという特徴がある．

そこで，同一ページでは固有名の前後は同じ区切り記号が使われやすいというヒューリスティックスを用いて，余分な部分の排除を行う．すなわち，既知の固有名の前後１文字を見て，新規の固有名もそれらの文字に囲まれている，と考えて余分な部分の排除を行う．

5.6　新規三つ組抽出

後処理を行って，獲得できた新規の三つ組を抽出する．

まず，パスによって得られた住所に関して地名データベースを参照し，住所としての正しさを判定し，かつ余分な部分を削除する．

固有名に関しては，前後１文字の判定でもうまく余分な部分が排除されなかったものなどを除くため，長すぎる文字列（25 文字以上）が抽出されてきていた場合は，その三つ組は無効であるものとする．また，パスによって示された場所に何も文字列が無かった場合も無効であるとする．

以上のチェックに引っかからなかったもののみを正しく抽出出来た三つ組として抽出する．

6　実験

本研究の手法を検証するために，小規模なデータを用いた実験を行う．

初期の三つ組のデータベースは，Yahoo!電話帳の中から神奈川県内の「グルメ・ドリンク」カテゴリの店舗のデータ 46,390 件を使用する．

6.1　評価基準

本研究の評価を行う上で，重要な点は４つある．

本研究では手法を繰り返し適用していくことで，三つ組のデータ数を増やしていく．新たな三つ組の獲得は, 新たな抽出対象ページの発見につながるだけではなく, 既に獲得済みのページからも以前見つからなかった構造を発見し，それによる三つ組抽出を行うことが出来る．Web 上に記述されている三つ組を網羅的に取るためには, 新しい抽出対象ページの発見と新しい構造の発見が双方とも効果的に機能している必要がある．そのため, この二点に対する評価は重要である．

新しい抽出対象ページの発見は, 抽出された三つ組からどれだけ新しいページが得られるかで評価し, 新しい構造の発見の評価は, 初期の三つ組だけで構造が発見されなかったページのうち, 抽出された三つ組を加えることで構造が発見されるようになったページがいくつあるかで評価する．

また，それと同時に抽出結果に対する評価も重要である．抽出された三つ組はそれ自体が結果としてデータベースに登録されるだけではなく，再帰的に用いられるため，不適当な三つ組を獲得してしまうとそれ以降の結果に悪影響を及ぼす可能性がある．同時に，抽出すべき三つ組が抽出出来ていない抽出漏れに関してもきちんとした評価を行う必要がある．

この抽出精度と抽出漏れに関しては, 収集されたページの中からランダムにサンプリングしたページを人手で確認して評価する．

6.2　実験方法

実験は次のように行う．

表 1: 実験結果

	一周目	二周目
データベース内三つ組数	46,390	88,932
全対象 Web ページ数	18,870	43,182
（うち、一周目残り）		18,521
電話番号が存在	18,161	41,827
電話番号が複数存在	7,311	18,634
HTML Tidy が適用可能	6,592	12,227
既知三つ組が複数存在	731	1,174
新規電話番号獲得	729	1,171
新規三つ組獲得	349	523
（うち、一周目残り）		79
抽出された三つ組数	42,542	45,001

1. データベース中の全ての三つ組を用いて抽出対象ページを検索
2. 1 で得られた全てのページに対して抽出手法を適用
3. 抽出された三つ組をデータベースに追加
4. 追加された三つ組を用いて抽出対象ページを検索
5. 4 で得られたページ ＋ 2 で三つ組が 1 つも抽出されなかったページに対して抽出手法を適用
6. 3〜5 を再帰的に行う

抽出対象ページ検索・抽出手法適用・データベース追加は, 同時に実行することが可能だが, 上で述べた新しく得られた三つ組による構造発見とページ発見を評価するために, 上のように段階的に手法を適用していく.

7 結果・考察

7.1 結果

実験の結果を表 1 に示す.

ここで, 一周目・二周目とはそれぞれ前節の実験方法における 1〜2,3〜5 にあたる.（うち、一周目残り）とは一周目で三つ組が一つも発見されず, 二周目で再度手法を適用したページ数と, それによって新たに三つ組を獲得することが出来たページ数である.

また, 本研究の手法は電話番号が複数存在している文書でなければ処理を行えないことから, プログラムの高速化のために, 始めにページ内に電話番号が複数存在しているかチェックしている.「電話番号が複数存在」の項目は, そのチェックに通ったページ数である.

「既知三つ組が複数存在」の項目は, 電話番号のチェックを通ったページの中で HTML Tidy が適用でき, 既知の三つ組が複数存在したページ数を示している.「新規電話番号獲得」は, その中からパスによって新規電話番号が獲得出来たページ数,「新規三つ組獲得」は, さらに完全パスを用いて固有名と住所の特定が出来たページ数を示している.

各周で抽出された三つ組のうち, 重複した組[9]を取り除いた数を「抽出された三つ組数」で示している.

[9]本研究では固有名・電話番号・住所の組み合わせが違うものは別の三つ組として数えている

データベース内三つ組数・抽出された三つ組数以外の数値は当てはまるページ数を示している.

一周目で抽出出来た三つ組に含まれる電話番号のうち, Yahoo!電話帳（地域・カテゴリ限定なし）に登録されていないものは 3,477 件あった.

7.2 考察

本手法を適用することで, 一周目・二周目ともに新しい三つ組を大量に抽出することが出来た.

前述した新しい抽出対象ページの発見に関しても, 2 周目で新たに 24,661 ものページを収集することが出来ており, 十分に効果的に機能していると言える. また, 一周目で構造が発見できなかったものに対して二周目で構造が発見できたページは 79 ページあった. この結果は, 本研究の再帰的な手法が効果的に機能した結果であると考えられる.

加えて, 一周目で抽出された三つ組のうち, 約 8 割が Yahoo!電話帳に登録されていなかったことから, Web から情報を抽出することで既存の住所録よりも網羅的なデータベースを作成することが出来るということを示している.

結果から, 既知三つ組を複数所持している Web ページの割合が少ないことがわかる. これは, 既知の固有名・住所がページ中にあるかどうかをデータベース中の固有名・住所との完全一致で行っていることに問題があるためだと考えられる. 固有名・住所のマッチングを DP マッチングなどの類似度ベースで行うことでより多くのページで既知三つ組を複数発見出来ると考えられるが, その場合抽出される三つ組の精度に影響がでてくる可能性がある.

また, 結果からは新規電話番号が抽出出来た Web ページ数に比べて新規三つ組が抽出出来た Web ページ数が少ないことにも気づくが, その理由については後述する.

7.3 評価

抽出精度と抽出漏れを評価するために, 収集した Web ページから以下のような評価セットを作成し, 人手で評価した.

- 三つ組が抽出されたページからランダムに 50 ページ（精度評価用）
- 既知の三つ組が複数発見され, かつ新規電話番号が発見されたページのうち, 三つ組が抽出されなかったページからランダムに 50 ページ（抽出漏れ評価用）

精度評価

抽出精度は, 抽出出来た三つ組のうち, 抽出元のページを人目で見て, 正しく抽出されたと判断されるものがいくつあるかで評価する. 抽出された電話番号が本当に使用可能であるかや, 住所が実在するかなどは評価しない.

評価対象の 50 ページから抽出された三つ組は全部で 2,594 組存在した. うち正しく抽出された三つ組は 2,226 組あり, 精度は 85.8%となった.

間違えて抽出されていたものの中で一番多かった原因は, 前後文字による固有名特定の失敗であった. 例え

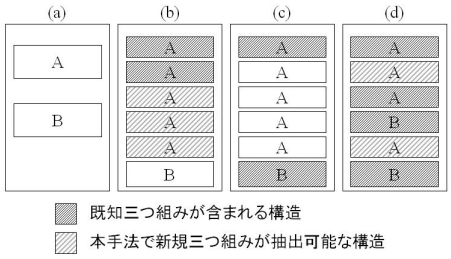

図 6: 抽出可能な構造

ば, 以下のように固有名が並んでいるページでは, 既知の固有名が 2 つとも行頭と空白で区切られており, そのルールで固有名を特定すると「有限会社」「居酒屋」などが固有名として獲得されてしまっていた.

- 既知固有名 ○○支店
- 既知固有名 ○○支店
- 有限会社 固有名
- 居酒屋 固有名

固有名や住所に対するパスの間違いや, 三つ組の組み合わせ違いなどのページの構造を間違って解析してしまったことが原因と思われる誤抽出は特に見つけられなかった.

抽出漏れ評価

抽出漏れの評価は, 三つ組が抽出されなかったページのうち, 三つ組が抽出されるべき構造を持つページがいくつあるかを評価することで行う.

この評価の際には, そのページで複数の三つ組が同じ構造で表記されているかが問題となる. しかし, 人が同じ構造であると判断できても機械的に判断することが難しい場合がある. 例えば, その他の部分は全て同じだが, 途中に小さな画像が挿入されている構造といない構造を同じ構造であると機械的に判断するのは非常に難しい.

本研究では, 上記のようにある構造に影響を及ぼすような要素が加えられた場合, その構造は元の構造とは別の構造と考える.

ここで, ページ中に三つ組を含む構造 A と構造 A にある要素が加えられた構造 B が存在するようなページを考えてみる（図 6）. このとき, 構造 A と構造 B が一回ずつしか出現しない場合 (図中の (a)) は, 本研究では抽出可能なページではないとする. また, 構造 A が複数回出現していても, 構造 B が一回しか出現していない場合は, 構造 B は抽出対象外とする. このうち, 図の (b) のように既知三つ組が構造 A で複数回記述されている場合は, 構造 A で記述される全ての構造が抽出可能であるが, (c) のように構造 A・B 双方で一回ずつしか記述されていない場合は, どちらの構造からも新規三つ組を抽出することは出来ない. ただし, (c) のような場合は手法を数回適用し, 構造 A で記述された三つ組をもう一つ発見することが出来れば, その時点で構造 A は抽出可能な構造と考えられる.

以上を踏まえた上で, 評価を行った. 評価の結果, 50 ページのうち, 三つ組が抽出されるべき構造を持つページは 36 ページも存在した. しかし, この 36 ページのうち 26 ページは (c) のようなパターンであり, 手法を再帰的に適用することによって三つ組が抽出されることが期待される. 残りの 10 ページは, 固有名・住所・電話番号を br タグ（改行）で区切ることで記述したページであった. 現在の手法ではこれらに対応することは出来ないが, 今後これらのページからも三つ組を抽出出来るようにしたい.

抽出不可能と判断した 14 ページは, 図 6 中の (a) のパターンのページや, 構造が全く認められないページ, またはページ中の構造から考えられる三つ組みと既知の三つ組の組み合わせが異なるページなどであった.

8 まとめ

本研究では，固有名への住所情報を含むメタデータ付与に用いることの出来る，固有名・電話番号・住所の三つ組のデータベースを Web 上から抽出することで生成する手法を提案した.

既存の住所録と比較した場合，Web 上からデータを抽出する利点として大規模なデータベースを構築出来る点，固有名の表記の揺れを獲得出来ることと，電話番号・住所以外のメタデータを獲得出来ることが挙げられるが, 本研究ではこれらの利点を十分に生かし切れる段階にまでは至っていない. しかし，上記の性質を生かしたデータベースを構築するための第一歩として重要な意味を持っている.

本研究の次のステップとしては, 抽出された三つ組のカテゴリ分けが考えられる. カテゴリはそれ自体が固有名に対するメタデータとなるだけでなく, 例えば飲食店ならば営業時間がメタデータとして存在しうるなど, その他にどのようなメタデータを抽出すべきかの指標となる. カテゴリの特定に関しては, 本稿で示したような抽出手法で実現することが難しいと考えられるため, カテゴリ特定に適した手法で行う必要がある.

参考文献

[1] 相良毅, 有川正俊, 高橋昭子. Xml を基本としたテキスト空間情報ベース. 情処研報, 99/61, pp. pp219–224, 1999.

[2] 関根聡. テキストからの情報抽出. 情報処理, Vol.40, No.4, pp. pp.370–373, 1999.

[3] The World Wide Web Consortium. Semantic web. URL: http://www.w3.org/2001/sw/, 2001.

[4] Project balog. 場 log. URL: http://www.balog.jp/, 2003-2004.

[5] 科学技術振興機構デジタルシティ研究センター. Universal design of digital city. URL: http://www.digitalcity.jst.go.jp/index-j.html.

[6] 佐藤理史. ワールドワイドウェブを利用した住所探索. 情報処理学会論文誌, Vol.42, No.1, pp. pp59–67, 2001.

[7] Google. Google web apis - home -. URL: http://www.google.com/apis/, 2004.

[8] Dave Raggett. Clean up your web pages with html tidy. http://www.w3.org/People/Raggett/tidy/, 1994-2003.

[9] The World Wide Web Consortium. Document object model. URL: http://www.w3.org/DOM/, 1997-2004.