データベースシステム 120－25
（2000．1．25）

ジオリファレンス情報を用いた空間情報媒介システム

相良 毅† 有川 正俊† 坂内 正夫‡

WWW ページやインターネットメールをはじめとするネットワーク上には，ジオリファレンス記述を含むデータが多数流通している．これらのデータには地名や住所が文章中に埋め込まれているため，効率よく取り出して緯度経度のような空間的位置に変換すると，ナビゲーションシステムなどの空間情報アプリケーションで利用できるようになり，高度な応用が可能である．本稿では，ネットワーク上を流通する情報に対してこのような変換処理を行う空間情報媒介システムを提案し，構成とアルゴリズムを説明する．また，プロトタイプシステム『芭蕉 Ver.2』を利用して変換精度と変換速度について実験を行い，考察を行う．

Spatial Information Mediation System using Geo-reference Description

SAGARA Takeshi† ARIKAWA Masatoshi† SAKAUCHI Masao‡

Today, a lot of data are delivered via computer network including WWW pages and Internet Mail, and fair part of them includes geo-reference description. Once they are extracted and converted to coordination as longitude and latitude, they become very useful for spatial information system applications like navigation. In this paper, this extraction and conversion system is named as "Spatial Information Mediation System" and its algorithm is proposed. Also, an experimentation result for conversion accuracy and efficiency is shown using prototype system "Basho version 2".

1. はじめに

WWW を利用した情報提供がますます盛んになっているが，その中で特に活発なものの一つに，地理情報の提供・交換がある．その要因として，携帯電話・携帯端末がインターネットに接続できるようになったため，出かけた先でネットワークから周囲の情報を収集する，いわゆる「人ナビ」に対する要求が高まったことが考えられる．このような要求にこたえるためには，実世界の現象や物体を高速に収集し，地図上に抽象化してデータベース化する技術が不可欠であるが，主に人手による情報入力に頼っているのが現状であり，金銭的・時間的コストが問題になっている[1].

†東京大学空間情報科学研究センター
Center for Spatial Information Science at the University of Tokyo
‡東京大学生産技術研究所
Institute of Industrial Science, University of Tokyo

そこで坂内らは，実世界での事象を撮影した静止画像や動画像を処理して抽出，地図データベース上に投入するシステムを提案し，実世界型情報媒介システムとして研究を進めている[2]．情報媒介システムは一種のメディエータであり，認識技術を用いて情報を抽象化して抽出するシステムを指している．静的な情報にたいする変換手法ではなく，連続して変化する動的な情報源から次々にデータを抽出する点に特徴がある．我々は，この仕組みを場所に関する記述（＝ジオリファレンス）を含むテキスト文書情報に適用し，空間情報媒介システムと呼ぶ．

空間情報媒介システムは，WWW ページ，ネットワーク上を流通するメールやニュースなどの情報や，文字放送，新聞記事などあらゆる情報から，ジオリファレンスを含むものを取り出し，空間的位置をキーとして管理するシステムを目指す．ジオリファレンスとは，地球上の位置を間接的に表現する文字列である．例えば住所，電話番号，郵便番号，交差点

名，駅名などがある（緯度経度など直接位置を表現したものはジオリファレンスとは呼ばない）．このシステムを利用することによって，これまで偶然にしか知り得なかった身の回りの情報を能動的に収集することができる．

本稿では，空間情報媒介システムのうち主にWWWページとインターネットメールを対象とし，住所と地名をキーとして場所を抽出する部分のプロトタイプシステムである『芭蕉 Ver.2』を紹介する．以下，2節でシステム構成と処理の流れについて説明し，3節で実装，4節で性能評価に関する検討を行う．

2. 空間情報媒介システムの構成

空間情報媒介システムは，図1のような一種のフィルタとして捉えることができる．入力はジオリファレンスを含む自然言語テキスト文書であり，特に空間情報として構造化されていないものでも構わない．例として，WWWページ文書やインターネットニュースの記事，インターネットメールなどが挙げられる．それ以外の文書でも問題はないが，情報媒介システムの目的は流通・変化する情報から有用なデータを抽出することにあるため，静的な文書を扱う場合は特に対象としない．言い換えれば，静的な文書であれば十分時間をかけて最適なインデックスを作成することができるため，情報媒介システムの利点があまり活かせない．

空間情報媒介システムの出力は，XML 表記法にしたがったSPA表現[3]による半構造化が行われた，XML 文書である．SPA 表現は，ジオリファレンス部分に XML タグを利用して地名語や緯度経度などの情報を埋め込んだもので，簡単な XML パーサを使用するだけで容易に位置情報とキーワードを取り出すことができる．位置をキーとした空間データベースに直接格納せずに SPA 表現を利用するのは，SPA表現を一種の中間表現として利用することで，その後に処理を追加しやすくするためである．例えば，利用者にとって興味のある空間的範囲内のデータだけを抽出するといった選択処理や，空間的属性以外のキーワードをさらに抽出するような処理が考えられる．図2は，「ラーメン」というキーワードを含むページ内の位置だけをポイントした例である．

空間情報媒介システムは，以下の4つの部分から構成される．

・文書構造解析
・形態素解析
・アドレスマッチング
・SPA 表現出力

2-1 処理の流れの概略

WWW 文書では HTML が利用されているので，HTML タグで文章が切断されてしまっている．そのため自然言語として不自然で，タグを取り除くなどの処理を行わないと次の処理である形態素解析の際に精度を下げる原因となってしまう．また，インターネットメールではメールヘッダやシグネチャの部分に Sender の住所のような余計なジオリファレンス情報が含まれていることがあり，誤った出力の原因になる．そこで，まず文書の特徴に合った文書構造解析を行う必要がある．

次に，ジオリファレンスの部分を文字列として正しく切り出すために，文書構造解析の結果得られた自然言語文章を形態素解析によって品詞分解する．切り出された部分文字列は，アドレスマッチング手法により地名辞書から検索し，一致するものがあれば空間的位置を返す．一致するものが無い場合はジオリファレンスではないと判断する．最後に，検索に成功した空間的位置を SPA 表現に直して元の文

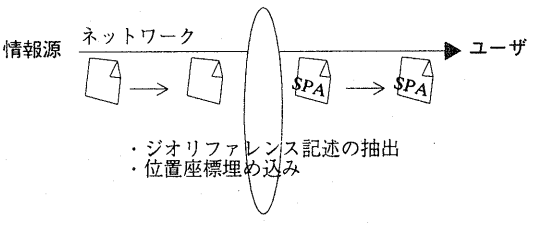

図1 空間情報媒介システム

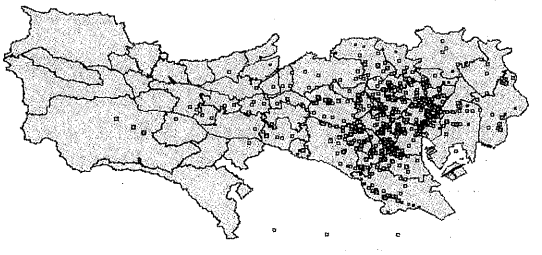

図2 WWW から抽出した東京のラーメン屋分布

書に埋め込み，出力する．

以下，各処理について詳しく説明する．

2-2 文書構造解析

将来的にはさまざまな文書に対して個々の解析アルゴリズムが必要だが，本稿では一例として，HTML文書とインターネットメールの構造解析を行うアルゴリズムについて説明する．HTML文書にはアンカーやフォントを指定するなどのHTMLタグが埋め込まれているため，そのままでは日本語として意味がとおらない．形態素解析では文章としての連続性がもっとも高くなるように品詞分解を行うため，タグで細かく切断された文章は形態素解析の際に誤る可能性が高くなってしまう．そこで，次の手順によってタグを取り除き，自然な文章を取り出す．

(1) HTML文書の先頭から，句点または改行を含む以下のタグが現れるまでを一文として取り出す
[P, H, HR, TABLE, TR, TD, UL, OL, LI, DL, DT, DD, BR, DIV, BLOCKQUOTE, PRE, CAPTION, FRAME]
(2) 文中から '<' と '>' で囲まれる部分文字列を探し，タグとして取り出す
(3) タグを取り除いた文を候補1として出力する
(4) 取り出した各々のタグに含まれる文章をそれぞれ候補2〜nとして出力する

ジオリファレンス記述はほとんどの場合 (3) で取り出した候補1に含まれるが，<IMG SRC= "略図画像名" ALT= "地名" >のようにタグ内に地名が含まれることがあるため，タグ内の文字列も取り出して検査する必要がある．

インターネットメールの場合は，まずメールヘッダ部分（最初の空行まで）を除去する．シグネチャ部も判断して除去する必要があるが，検討課題である．また，近年インターネットメール本文にHTMLが利用されている場合があるため，メールヘッダを除去した後で上述のHTML文書構造解析を行う．

2-3 形態素解析

形態素解析モジュールでは，形態素解析システム『茶筅』[4]を利用して品詞分解を行い，分解された単語列から「地名らしい」ものを選択する．形態素解析では辞書を利用して「もっとも連続性の高い品詞列の組み合わせ」を求めるため，辞書に登録されていない地名は正しく切り出されない．多くの場合，地名は複合名詞になっているため，一語であるはずの地名が複数の連続する名詞に分解されてしまう．この点を考慮し，「地名らしい」単語列の選択基準として，以下のルールを適用する．

ルール1: グループ1に含まれる品詞から始まる単語列であり，かつ
ルール2: グループ1またはグループ2に含まれる単語が0回以上連続していること

グループ1の品詞

名詞・固有名詞・一般，名詞・固有名詞・人名・一般，名詞・固有名詞・人名・姓，名詞・固有名詞・一般・名，名詞・固有名詞・組織，名詞・固有名詞・地域・一般

グループ2の品詞

名詞・一般，名詞・固有名詞・人名・名，名詞・固有名詞・国，名詞・サ変接続，名詞・数，名詞・接尾・一般

（品詞区分は『茶筅』に従う）

このルールは，現時点でアドレスマッチング用の地名辞書に登録してある地名（表1）が当てはまるように調整した最小の組み合わせであり，登録されていない地名の中にはこのルールで選択されないものがある可能性がある．グループ1, 2により多くの品詞を加えれば，それだけ地名を拾う可能性は高くなる（再現率が高くなる）が，地名以外の文字列を拾う可能性も高くなり（適合率が低くなる），処理速度が著しく低下する．そのためアドレスマッチング辞書を拡張する場合は，辞書にあわせてルールを調整する必要がある．

2-4 アドレスマッチング

アドレスマッチングモジュールは，形態素解析モジュールが「地名らしい」と判断した単語列を文字列として受け取り，地名辞書を参照する．辞書中に存在すれば，該当するレコードに含まれる空間的位置（緯度経度座標）を返し，存在しない場合は地名ではないと判断する．

地名は多くの場合階層構造を持っているが（東京大学／空間情報科学研究センター，東京都／目黒区，小田急線／下北沢駅など），自然言語で表現すると上位階層が省略される場合がある．省略に対応するため，木構造で地名を階層的に管理し，さらにそれぞ

れのノードに対して辞書インデックスを持つ二重管理構造を利用する（図3）．形態素解析モジュールから渡された文字列を，この管理構造を利用して検索する手順を以下に示す．

(1) 辞書インデックスを利用して，文字列の先頭から一致文字数が最長となるノード N_{1max} を見つける

(2) ノード N_{1max} の子ノード集合 $N_{2(1..m)}$ （m は子ノード数）から，文字列の続きの部分が一致するものを探し，親ノードを含めた一致文字数が最長となるノード N_{2max} を見つける

(3) （2）を再帰的に繰り返し，最長一致ノード系列 {Nimax}（i=1..n, n はノード列長）を得る

(4) step4: N_{1max} の親ノードを上に辿り，上位階層も含んだ完全なノード系列 {Nmax} を得る（（1）～（3）において，最長一致解が複数ある場合はその全てについて同じ処理を行う）

地名には同名のものがあるため，解が複数存在することがある．そこで，一致度に応じて次のような得点付けを行い，得点が高いものから採用する[5].

2-5. SPA 文書出力モジュール

アドレスマッチングの結果，最長一致ノード系列 {Nmax} と代表点の座標が得られる．この情報をSPA 表現に変換し，元の文書中に埋め込んで出力する．

3. 空間情報媒介システムの実装

2 節で説明した 4 つのモジュールを，UNIX 上のプログラムとして実装した．文書構文解析モジュール部は Perl で記述し，HTML 文書とインターネットメールの構文を整形する．形態素解析モジュールも Perl で記述したが，実質的な処理は外部プログラムとして呼び出される『茶筅』が行う．

内容	備考	件数
全国の市町村名		4118 件
住所データ	東京都，埼玉県，千葉県，神奈川県	618,156 件 重複あり
首都圏駅名	数値地図 2500 より	1272 件
首都圏建物名	数値地図 2500 より	37803 件
交差点名，駅名，高速道路インターチェンジ名など	全国デジタル道路地図中より	24921 件

表1 アドレスマッチングテーブルに登録済みデータ

アドレスマッチングモジュールはクライアント・サーバ形式のプログラムとして実装した．クライアント側は Perl で記述し，検索したい文字列をサーバに送り，検索結果を受け取る．サーバ側は C++で記述し，デーモンプロセスとしてクライアントの要求を常時受け付ける．今回の実装では検索速度向上のため try 検索木を辞書インデックスに利用している．SPA 文書出力モジュールは Perl で記述した．

また，サーバ側のアドレスマッチングモジュールを除いた部分は，UNIX のフィルタプログラム『芭蕉 Ver.2』として実装した．標準入力からジオリファレンスを含む文書を与えると，SPA 表現を含んだ文書を標準出力から出力する．例えばメールを受信する際にこのフィルタプログラムを通すように設定すると，受信するたびに緯度経度情報が付加されるので，空間情報媒介システムとして利用できる．

4. 実験・考察

本プロトタイプシステムの性能を評価するため実験を行った．49 個の地名を含むサンプルテキスト文書を与え，SPA 文書に変換した（図4）．ここでは，該当レコードが 1 件のみ，かつ，文字列が見出し語と完全一致するものだけを変換しているので，駅の名前である「恵比寿」や「渋谷」，建物名として登録されている「西麻布郵便局」や「防衛庁」，及び「六本木 5-9-22」のような住所が抽出されて SPA 表現に置き換えられており，十分空間的な処理に利用できるといえる．しかし，人手でチェックした結果，次

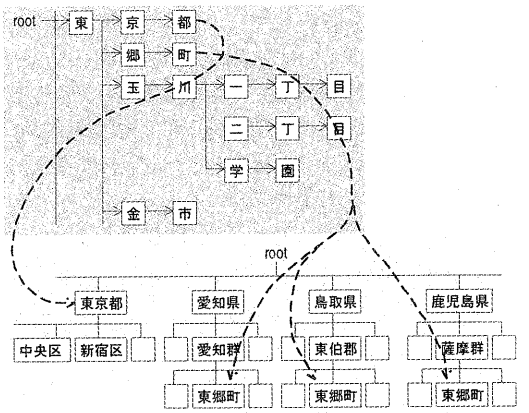

図3 アドレスマッチング辞書の二重管理構造

の2つの問題があることが分かった.

ケース1)「六本木」が変換されていない

「六本木」は「渋谷」同様,駅の名前なので変換されるべきだが,変換されなかった.これはデジタル道路地図に岩手県東和町の「六本木」が含まれていたため,該当レコードが2件になってしまっているためである.

変換されなかったのはアルゴリズム的には「正しい」と言えるが,地名辞書が充実するほど変換しなくなる確率が高くなることになり,なんらかの解決策が必要である.例えば文中の前後を見て,どちらの「六本木」を意味しているのか推測するといった方法が考えられる.

ケース2)「防衛庁」が一度目には変換されず,二度目には変換されている

変換されなかった一度目は「駅から防衛庁へ向かい手前を右へ」という文中に現れているが,これを『茶筅』で変換すると,「防衛庁」の前にある格助詞の「から」との接続関係により,「防衛」(名詞・サ変接続)+「庁」(名詞・接尾・一般)に分割されている.形態素解析モジュールでの地名語抽出ルールで,「名詞・サ変接続」がグループ1に含まれていないため,このように分割されていると抽出できない.一方,変換できた二度目は「防衛庁向かいのビルの中」という文に現れており,「防衛庁」(名詞・固有名詞・組織)という一語に分解されているため抽出できる.

この問題は,地名語抽出ルールのグループ1に「名詞・サ変接続」を加えれば解決する.しかし,辞書からルールを作成した際にはこのような問題は推測できなかったため,ほかにも同様の問題が起こる可能性があり,より確実なルールの検討が必要である.

次に,『芭蕉 Ver.2』の速度性能を調べるため,この変換に要した時間をtimeコマンドで計測した.実験に利用した環境はSUN Enterprise450 (2CPU)である.計測結果は32.00秒であり,これは同じマシン上で前バージョンの『芭蕉 Ver.1』[3]を利用して同じファイルを変換するのにかかった3:29.46秒に比べ6.54倍向上している.主な理由として,1)文書構文解析モジュールの採用,2)アドレスマッチングで階層構造を採用したことの2点を考え,検証を行った.

Ver.1では文書構文解析を行っていないため,HTMLタグや改行で文章が細かく切断されている.そのため本来一つであるはずの文を処理するために複数回『茶筅』を呼び出してしまう.形態素解析では接続関係による制約があるので,文が長くなっても処理速度はあまり増加しない.そのため,文書構文解析モジュールによって文をつなげば高速になるはずである.そこで,文書構文解析モジュールの機能をオフにして同じ実験を行ったところ,38.16秒かかった.劇的な改善効果は見られないが,これは構文解析自体にかかる時間で相殺されている可能性がある.

アドレスマッチング部は,Ver.1ではRDBを利用していたため,階層構造の管理効率が非常に低いという問題があった.一方Ver.2では管理構造をすべてC++で書下ろし,ディスク上に木構造を構築しているため,階層構造の検索が高速になった.この効果を確認するため,100個の住所テーブルを直接アドレスマッチングモジュールに送り,変換に要する時間を測定した.Ver.1のRDBを利用したアドレスマッチングでは1486秒,Ver.2では245秒であり,約6倍速度が向上しており,性能向上が確認できた.

空間情報媒介システムとしての有効性という点からは,1ファイルの変換に32秒という性能は,1時間に100ファイル以上を変換できるということであり,受信したインターネットメールをすべて変換するには十分な性能であるといえる.しかし,WWWブラウズの際に利用することを考えると毎回30秒も待たされるのは実用的とは言えず,更なる改善が必要である.

5.おわりに

本稿では,空間情報媒介システムを提案し,そのプロトタイプシステム『芭蕉 Ver.2』を紹介した.本プロトタイプシステムは,WWWページやインターネットメール中のジオリファレンス記述を自動的に見つけ,空間的位置に変換することができる.

また,変換精度と速度についても実験を行った.精度については,実用上十分な結果が得られたが,今後さらに精度を向上するために,文の前後やページのリンク関係など潜在的な空間的範囲の制限を考慮する必要がある.速度面では,ある程度情報伝達に遅れが生じても構わないインターネットメールでの利用には十分な性能を示した.しかし,情報媒介

システムとして，WWW ブラウズなどインタラクティブな利用をするにはまだ不十分であり，さらに高速化を進める必要がある．

参考文献

[1]三浦 信幸，横道 誠司，高橋 克巳，島 健一，“GIS を用いた位置指向の WWW サーチエンジン～モーバイルインフォ 2 実験～”，地理情報システム学会講演論文集, Vol.7, pp.131-136, 1998

[2]Haomin Jin, Masao Sakauchi, “The Automatic Selection and Recognition of Objective Parts in Real World Building Image”, IAPR workshop on machine vision applications 98, pp339-342, 1998

[3]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 Vol.99, No.61, pp.219-224, 1999

[4]松本 裕治，北内 啓，山下 達雄，今 一修，今村 友明，“日本語形態素解析システム『茶筅』Version1.0 使用説明書”，NAIST Technical Report, NAIST-IS-TR97007, 1997 (http://cactus.aist-nara.ac.jp/lab/nlt/chasen.html)

[5]相良 毅，有川 正俊，坂内 正夫，“ネットワーク上各種情報源からの地理情報抽出収集手法”，地理情報システム学会講演論文集, Vol.8, pp.331-334, 1999


```
<HTML>
</HEAD>
<TITLE>
東京の<spa><name>東京</name><geoword>東京
</geoword><longtude>35.677608</longtude
><latitude>139.769257</latitude><domain
>東京</domain><style><color>yellow-
</color><size>3</size></style><author>-
BASHO Ver.2.1 with DAMS</author></spa>ラ
ーメン屋さん</TITLE>
<HEAD>
<BODY BACKGROUND="../img/back3.gif" BODY
LINK=#710502 BODY VLINK=#135797>
……<中略>……
<A NAME="ropponngi">
■六本木</A>  ←六本木が変換されていない
</FONT>
……<中略>……
<FONT SIZE=6>
赤のれん</FONT>
  六本木  西麻布 3-21-24(<spa><name>東京都港区西
麻布3丁目21</name><geoword>西麻布 3-21-
24</geoword><longtude>35.656212</longtude><
latitude>139.728195</latitude><domain>東京都
港区西麻布3丁目21</domain><style><color>-
```

```
red</color><size>4</size></style><auth
or>BASHO Ver.2.1 with DAMS</author>-
</spa>246 を渋谷方面へ<spa><name>渋谷
</name><geoword>渋谷方面</geoword>-
<longtude>35.655849</longtude>-
<latitude>139.704575</latitude>-
<domain>渋谷</domain><style><color>-
yellow</color><size>3</size></style><a
uthor>BASHO Ver.2.1 with DAMS-
</author></spa>外苑西通り手前左側）<BR>
ラーメン  650  11:30～21:00 日祝<BR>
……<中略>……
size></style><author>BASHO Ver.2.1 with
DAMS</author></spa>駅から防衛庁へ向かい手
前を右へ）  <BR>  ←防衛庁が変換されていない
……<中略>……
防衛庁向かいの<spa><name>防衛庁</name>-
<geoword>防衛庁向かい</geoword>-
<longtude>35.663139</longtude>-
<latitude>139.734161</latitude>-
<domain>防衛庁</domain><style><color>-
yellow</color><size>3</size></style>-
<author>BASHO Ver.2.1 with DAMS-
</author></spa>ビルの中）<BR>
                    ←防衛庁が変換されている
```

図 4 芭蕉 Ver.2 による変換結果