特　集　「テキストマイニング」

Webマイニング

Web Mining

坂本 比呂志　九州大学大学院システム情報科学研究院
Hiroshi Sakamoto　Department of Informatics, Kyushu University.
hiroshi@i.kyushu-u.ac.jp

有村　博紀　九州大学大学院システム情報科学研究院＆科学技術振興事業団さきがけ研究21
Hiroki Arimura　Department of Informatics, Kyushu University & PRESTO, Japan Science and Technology Corporation.
arim@i.kyushu-u.ac.jp

Keywords: World Wide Web, semi-structured data, information extraction, data mining, machine learning.

1. はじめに

インターネットの急速な普及は，社会を大きく変えつつある．とくに，Webページと Web 検索エンジンの利用の広がりとともに，社会・経済・科学に関する多様かつ大量の情報が，HTML や PDF で記述された Web ページやデータシート・オンラインデータベースといったさまざまな形態で，インターネット上に集積されている [Lawrence 99b]．また，今後10年間には，科学と経済に関するより多くの情報が，Web 上で無償で公開され，広く利用できるようになると予測されている [Lawrence 99a].

現在，これらのインターネット上に蓄積されたデータの多くは，人間である利用者が直接見たり読んだりするためのものであり，計算機が自動的にこれらのデータからデータにアクセスし，情報を自動的に取り出すためのものではない．しかし，将来は，計算機の支援のもとに，これらの大量のデータを解析して，人間である利用者に提示したり，データ書式に関するゆるい合意のもとに，計算機どうしが情報を交換したりすることも考えられている．そのため，これらのインターネット上に蓄積された大量の情報から，計算機を用いて，有用な情報を取り出すための技術が必要となっている．

本稿では，これらの技術を，広い意味で Web マイニングと呼ぶことにする．これらの技術は，Web 上の情報源の探索 [Ambite 00]から，ユーザーのモデリング，情報の自動抽出 [Kushmerick 00]，Web 知識ベース [Craven 00]，文書の自動分類 [Schapire 00] にいたるさまざまな問題を対象としており，同時にプランニング，分散データベース，機械学習などの広い範囲の技術に基づいている．

Web マイニングは確定した分野とはなっていないため，これが Web マイニングだといえる包括的な解説は難しい．そこで，筆者が最近興味をもっている Web データからの情報獲得と，半構造データからのデータマイニングに関する研究を中心に，いくつかの観点から Web マイニングの方向性をみていきたい．詳細については，それぞれの研究の参考文献を参照いただければ幸いである．また，自動文書分類に関しては，本特集の解説 [那須川 01] や最近の解説 [永田 01] をご覧いただきたい．

2. Web からの情報抽出

現在，インターネット上の多くの情報は HTML 言語で記述された Web ページの形で公開されている．そのため，与えられた Web ページの集合から，決まった書式の情報を取り出す情報抽出（Information Extraction）の研究が盛んに行われている．

2·1　Kushmerick のラッパー帰納問題

[Kushmerick 00] は，情報抽出をラッパー帰納問題の枠組みで詳細に調べた．情報抽出に関する従来の多くの研究は，対象となる特定の種類の HTML ファイルの構造に依存した経験則を用いた抽出が中心であったのに対して，Kushmerick は，情報抽出問題を，ラッパー帰納として明確に定式化することから始めた．

彼が定義した最も基本的なラッパーが，LR ラッパー（left-right wrapper）である．例として，図1に示した国番号データベースの簡単な HTML ファイルから，図2に示した対応する国名と国番号の対をすべて切り出す仕事を考えよう．[Kushmerick 00] は，もとの HTML

```
P =  <HTML> <TITLE> </TITLE> <BODY>
     <B> Congo </B> <I> 242 </I> <BR>
     <B> Egypt </B> <I> 20 </I> <BR>
     <B> Belize </B> <I> 501 </I> <BR>
     <B> Spain </B> <I> 34 </I> <BR>
     </BODY> </HTML>
```

図1　国名と国番号の HTML ファイル

$$L = \left\{ \begin{array}{l} (\text{‘Congo’;‘242’}), \\ (\text{‘Egypt’;‘20’}), \\ (\text{‘Belize’;‘501’}), \\ (\text{‘Spain’;‘34’}), \end{array} \right\}$$

図2 切り出された国名と国番号の属性レコード集合

$$L = \left\{ \begin{array}{l} ((50,55),(63,66)), \\ ((78,83),(91,93)), \\ ((105,111),(119,122)), \\ ((134,139),(147,149)), \end{array} \right\}$$

図3 帰納アルゴリズムの入力となる属性レコード集合

ファイル（図1）のことをページと呼び，切り出された属性レコードの集合（図2）のことをラベルと呼んでいる．この例では，各属性レコードは，国名と国番号の二つの属性を含む．

さて，図1に示されたページを観察すると，各行で <B> と </B> の一組のタグにはさまれた部分テキストが，国名に対応しており，それにつづく <I> と </I> の一組のタグにはさまれた部分テキストが，国番号に対応していることがわかる．そこで，ラッパーとしては2組のタグからなる組

(('<B>', '</B>'), ('<I>', '</I>'))

を，属性切出しのための区切り語ベクトルとして指定する．

区切り語ベクトルが与えられると，LRラッパーは次のように切出しを行う．まず，最初の属性に対する区切り語の組（'<B>', '</B>'）をみて，入力ページ（HTMLファイル）を先頭からスキャンし，左の区切り語'<B>'の最左出現と，それにつづく，右の区切り語'</B>'の最左出現を見つける．そして，左と右の区切り語ではさまれた部分テキスト（この場合はページPの2行目の'Congo'）を，その属性の値とする．これを次の属性についても繰り返して，一つの属性レコードを切り出す．さらに，これを可能なかぎり繰り返して，ページに含まれるすべての属性レコードを切り出す．

LRラッパーは，一般に，このような区切り語ベクトル$V=((l_1, r_1), ..., (l_k, r_k))(k \geq 1)$で定められるラッパーである．ラッパー$W$が，入力ページ$P$から出力である属性レコード集合$L$を切り出すとき，$W(P)=L$と書く．

ラッパー帰納問題（Wrapper induction）は，切出しの逆問題として定式化される．帰納アルゴリズムは，入力として，切出し対象のページPとそこから切り出したい属性レコード集合（ラベル）Lの組$(P_i, L_i)_{i=1}^{N}$を受け取り，定められたラッパーのクラス$\boldsymbol{W}$から，正しく切出しを行うような，すなわちすべての$1 \leq i \leq N$に対して$W(P)=L$となる，ラッパー$W \in \boldsymbol{W}$があれば，それを出力する．

彼は，ラッパー帰納問題を扱いやすくするために，帰納アルゴリズムに，各属性値の値だけでなく，入力ページのどの位置の文字列が属性値として切り出されたかという，位置情報を与えることにした．図3は，図2に対応する属性レコード集合の例である．この例では，属性値'Congo'は，ページにおける開始位置と終了位置の組（50, 55）で表現されている．

この種の情報の仮定は，一見不自然に見えるが，実際のシステムでは，利用者が適当なGUIを用いて位置情報を与えることは難しくない．一方，この位置情報の利用により，帰納問題は著しく簡単になる．例えば，第一属性の左区切り語（右区切り語）を見つけるには，各属性値'Congo', 'Egypt', 'Belize', 'Spain'の出現の左側で終わる（右側で始まる）文字列の最長共通接尾辞（最長共通接頭辞）を取ればよいことが示せる．このアイディアを用いて，Kushmerickは，無矛盾なLRラッパーを計算する効率良いアルゴリズムを与えた．

彼は，実際のデータに表れるさまざまなタイプの情報抽出問題に対して，LRラッパーのほかにも，HLRT（header-tailer LR），OCLR（open-close LR），HOCLRT（header-tailer open-close LR）ラッパーといったクラスを定義し，これらのクラスに対する，ラッパー帰納問題を解く効率良いアルゴリズムを与えた．またこれらのアルゴリズムの計算複雑さや，クラスの学習可能性についても詳細に調べた．また，CGIスクリプトによって生成されるオンラインデータベースをもつ，30か所以上の実際のWebサイトを対象にラッパー帰納実験を行い，帰納的に合成されたラッパーの有効性を検証している．

2・2 木構造上のラッパー帰納問題

[Kushmerick 00]の研究のほかにも，ラッパー帰納問題に関するさまざまなアプローチが行われている．筆者らによるTreeWrapperの研究を紹介する．

Kushmerickのラッパーは，HTMLテキストを，タグも含めてASCII文字上の単なる文字列とみなして，変換を行うものであり，そのために，比較的単純な変換規則で強力な書換えが行える．その一方で，タグ付きテキストのタグ構造を無視するため，ある種の変換は記述できないことがある．

そこで，[Sakamoto 01]では，HTMLによるWebページやXMLデータを，タグの入れ子構造によって定まる木構造と考えて，木構造からのデータの抽出を扱うラッパーTreeWrapperを提案している．タグ付きテキストは，文書全体に対応する根をもつ，辺にタグがラベル付けされ，節点にテキストがラベル付けされた木とみなせる[Abiteboul 00]．

基本的なアイディアは，切り出す部分構造の指定にパ

ス表現（path expression）を用いることである．パス表現とは，'paper'.*.'author'.'last_name' のようなタグ名の列であり，途中に任意のタグ名の有限列を表すワイルドカード * を0個以上含むことができる．例えば，上のパス表現は「'paper' タグでラベル付けされた辺から，いくつか辺をたどり，連続した 'author' と 'last_name' の下にぶら下がっている節点」を表す．一般に，タグ表現は，それを具体化して得られるタグ列を，根からたどって得られる節点（とその部分木）を表す．

切り出すレコードの属性数を $k \geq 1$ とすると，一般に，TreeWrapper は，パス表現の組 $(P_1, ..., P_k)$ で指定される．ここに，各 P_i はタグ上のパス表現である．与えられた HTML または XML ファイルに対応するラベル付き木 T を考えると，TreeWrapper は，各属性 $1 \leq i \leq k$ に対して T の根からパス表現 P_i で到達可能な節点を見つけ，その下の部分木全体に対応するテキストを属性値として切り出す．直感的には，TreeWrapper は，[Kushmerick 00] の LR ラッパーを，木のパスをあつかうよう拡張したものである．TreeWrapper に対しても，Kushmerick の学習法を拡張することができ，特殊な場合にラッパー帰納アルゴリズムを与えている [Sakamoto 00].

文献 [Sakamoto 00] と [Arimura 00b] では，半構造データのラッパー帰納問題を，ノード書換え系や項書換え系の例からの学習問題として定式化し，その複雑さを調べている．

3. Web 知識ベース

近い将来に，科学技術や社会・経済に関するさまざまな情報の多くが，Web 上に集積され，一般に利用されるようになると予測されている [Lawrence 99a]．現在の Web 検索エンジンは，与えられたキーワードと適当なランキング戦略に基づいて，利用者が望む Web ページの集合を見つけてくれる．しかし，Web ページの内容に関する質問に答えることはできない．野心的な試みとして，これを一歩進めて，Web ページ全体を知識ベースとして利用することを目指した Web 知識ベースの研究が行われている．

3·1 Web → KB プロジェクト

CMU の Tom Mitchell の研究グループは，Web → KB プロジェクトで，さまざまな Web ページから，知識ベースを自動的に構築する研究を行っている [Craven 00].

以下では，論文 [Craven 00] にあげている，大学の学科 Web サイトから知識ベースを自動構築する例を用いて説明する．Web → KB では，記述論理（description logic）に似た表現で知識ベースを表現する．構築する知識ベースは，対象に含まれる個体（object）に対して，それが属するクラス（class）と，個体間の関係名（relation）を記述するものである．

```
Jim's Home page
    I teach several courses:
      • Fundamentals of CS
      • Introduction to AI
    My research includes
      • Intelligent web agents
      • Human computer interaction
```

図4 Web → KB システムの入力となるホームページの例

```
jim ∈ faculty
    courses_taught_by:
      fundamentals_of_CS, introduction_to_AI
    home_page:
      http://www.dcs.cmu.edu/~jim/
```

図5 Web → KB システムの出力となる知識ベース項目の例

例えば，図4のような教官のホームページが，学科の Web サイトにあったとしよう．Web → KB では，各個体に，実際の Web ページのその個体を記述するテキストの全体または一部分が対応すると考える．したがって，このホームページ全体は，'faculty' クラスに属する 'jim' という個体に対応する．このホームページから，個体 'jim' を記述した図5のようなレコードを取り出して，知識ベースに追加するのが WEB → KB システムの目的である．

この図4と図5の例では，Web → KB は，このホームページ全体がクラス 'faculty' に属する個体を記述していると判断した．そこで，このページをクラス 'faculty' のクラス例という．さらに，図4のホームページの "I teach several courses..." という部分のテキストから，'courses_taught_by:' という2項関係の具体例を抽出し，このホームページ自体とその URL から，別の2項関係 'home_page:' を抽出している．

それでは，Web → KB は，どのようにして知識を抽出するのだろうか？ Web → KB では，あらかじめ知識ベースの構築対象となる分野に対して，知識ベースで用いるクラス名と関係名を，語彙（vocabulary, ontology）として人間が与えておく．図6は，学科 Web サイトにおける語彙の例である．

最初の3行は，クラスに関する記述であり，'person' は 'faculty' と 'staff', 'student' からなるというクラス間の階層構造を記述している．次の3行は，関係に関する記述であり，Web ページの 'faculty' を表す部分は，2項関係 'course_taught_by:' や 'student_of:' に関するテキストを含んでいることを記述している．

次に，語彙の各クラスに対して，与えられた Web ページ全体（または一部）の記述する個体が，あるクラスに属するかを判定する分類規則を，人間が与えた訓練例から学習する．訓練例は，Web ページと，そのページ

```
entity::= {activity, person, other}
activity::= {research_project, course}
person::= {faculty, staff, student}
entity: 'home_page:'
faculty: 'course_taught_by:' , 'student_of:'
course: 'instructors_of:'
```

図6 Web→KBシステムにおける語彙の例

から人間が切り出したクラス例の組である．クラスの学習のための機械学習手法として，Webページを英単語の属性空間モデル（bag of words model）ととらえて，一種の素朴ベイズ法（Naive Bayes）と一階論理式の関係学習法を用いている [Craven 00]．関係の学習には，ページのクラス，ページからのリンク，ページ中の単語を用いた関係学習法を用いている [Richards 92]．

実験 [Craven 00] では，クラスと関係に対する訓練例として，それぞれ，四つの大学の計算機学科のWebサイトから，8 000Webページと1 400Webページ対を訓練例として用いてシステムを訓練し，テスト例として選んだ別の大学（CMU）の計算機学科のWebサイトに適用している．始めに，各Webサイトの入り口ページから開始し，幅優先探索を用いて，そのサイトのWebページを探索していく．ページを訪れるごとに，訓練した分類アルゴリズムを用いて，それを語彙中のクラスの一つに分類し，さらに，ページに含まれる関係の具体例を抽出し，知識ベースに加える．分類できないときは，そのページから先の探索は中止する．実験の結果，2 722のWebページを訪問し，そのうち374の新しいクラスの具体例を知識ベースに追加した．抽出精度については，ほとんどのクラスと関係について70％程度以上だったと報告されている [Craven 00]．

3·2 WHIRL：類似性結合による情報統合

ここで紹介したWeb→KBと異なるアプローチとして，次のような研究がある．[Cohen 98] のWHIRLシステムは，利用者のさまざまな質問に対して，インターネット上に分散した異種Webサイトの情報を統合して回答する．WHIRLは，テキストデータを属性値として扱う一種の関係データベースであり，二つのテキスト間の類似度を用いて，異なるWebサイトを統合する．類似度は，例えば，二つの文書間のコサイン測度として定義する．例として，映画の上映館と時間に関するWebサイト‘movieListing’と映画評論に関するWebサイト‘review’があり，それぞれに対して，適当なラッパーが利用できるとする．ただし，映画の題名の表記法は“Men in Black”と“Men in Black, 1997”のように異なっているとする．このとき，二つの映画館の情報を結合して検索するのに，WHIRLでは次のような質問を使える．

? - movieListing(Cinema, Movie1, Times)
∧ review(Movie2, Review) ∧ Movie1 ～ Movie2.

あるいは，宇宙人に関するコメディを探すのに，次の質問ができる．

? - review(Movie, Review)
∧ Review ～ “comedy with space aliens”.

さらに，[Cohen 00] は，WebページのHTMLタグで定まる構造を，WHIRLシステムで扱えるように拡張し，先の類似度による検索とランキング機構を用いて，WHIRLがラッパーの自動構築も行えることを示している．

3·3 Webデータ上の引用データベース

Webマイニングとはいえないかもしれないが，情報抽出の実働システムと，その可能性を示唆する例として，[Lawrence 99b] らの引用データベースResearchIndex（Citeseers）を紹介する．ResearchIndexは，インターネット上に公開されている技術報告や論文原稿をWebロボット（crawler）によって収集し，それらに対する検索を提供するWeb検索エンジンである．

従来の専門的サーチエンジンと異なる点は，Research Indexが，ポストスクリプトファイルやPDFファイルの形式で公開されている論文を収集対象としている点である．さらに，これらのファイルからテキストを取り出して，全文索引をつくるだけでなく，内容と書式を解析して，論文の引用情報や書誌情報を抽出し，それを検索情報として公開している点である．開発者のLawrenceとGalesは，Web全体のサイズの見積もりに関する論文 [Lawrence 99a] で有名だが，もともとはニューラルネットワークの研究者であり，引用文献の解析には経験的手法と機械学習手法を併用している [Lawrence 99b]．

引用情報は，学術雑誌や会議録に掲載されている一般の論文の情報を含むため，無償で公開されているデータを用いて，質と量の揃った2次文献索引を構築することに成功している．著者による自己引用は区別するなど，なかなかよくできている．また，引用情報が充実しているため，Citation Indexの手軽な代用として用いている読者も多いのではないだろうか．

4. 最適化パターン発見を用いたWebマイニング

最後に，筆者自身の研究について紹介する．ネットワーク上に分散したWebページに代表されるテキストデータの利用が急速に進むにつれ，これらのテキストデータベースに対して，従来からの情報検索手法を超える新しいアクセス手法の開発が急務となっている．

そのための有力な方法が，データマイニング（Data Mining）である．しかし，(i) 明示的な構造をもたない (ii) 非均質で多様な文書の (iii) 膨大な量の集積である大規模テキストデータベースに対して，関係データベースを対象としている現在のデータマイニング手法を，そ

のまま適用することはできない.

そこで，筆者らの研究グループでは，テキストデータベースを対象とした新しいデータマイニング手法を開発し，実際の大量データに適用可能な効率良いシステムの実現方法を明らかにすることを目標として研究を行ってきた．とくに，現在のテキストアルゴリズムの理論と計算学習理論の成果を積極的に援用し，徹底して頑健かつ高速なパターン発見手法を開発することに重点をおいて，研究を進めてきた.

4·1 近接語相関パターンと最適化データマイニング

本研究では，パターンの族として，**近接語相関パターン**（proximity word-association pattern）と呼ばれる単純なパターンの族を採用した．順序あり近接語相関パターンとは，与えられたテキストの d 個の文字列（以後，フレーズまたは語と呼ぶ）のならびと，非負整数 k（近接度と呼ぶ）からなるパターン $\pi = (p_1, ..., p_d; k)$ である．例えば，< CACA, AGGAGG, TATA; 30 > は順序あり近接語相関パターンの例である．この順序あり近接語相関パターンは，与えられたテキスト中に，文字列 CACA の後に，AGGAGG および TATA が，互いに 30 文字以下の距離しか離れず，定められた順序で連続して出現するという制約を表現している．また，語順の制約を除き，フレーズが互いに近接して任意の順で出現するものを，順序なし近接語相関パターンと呼ぶ.

パターン発見の枠組みとしては，正負例の有限集合 $S \subseteq \sum^* \times \{0, 1\}$ が与えられたとき，例に対する分類誤差（empirical error）$Error_S(H) = \sum_{<x, b> \in S} [H(x) \neq b]$. を最小化するようなパターン H を見つける問題（**最適パターン発見問題**）を考察する．さらに，分類誤差を一般化して，頻度・確信度の最適化 [Fukuda 96] や，情報エントロピーや Gini 指標などの不均衡関数（inpurity function）の最適化にも適用する．例えば，分類誤差に関する最適パターン発見問題に関しては，最近の数理統計学と計算論的学習理論の研究から，この最適化問題を解くアルゴリズムは，データ中の雑音にきわめて頑健であり，未知パターンのクラスがわからない場合にも未知パターンをうまく近似することがわかっている．最近，データマイニングにおいてもこの枠組みの有効性が示されている.

4·2 高速な最適パターン発見アルゴリズム

パターンが含むフレーズの最大数を定数 d で制限したとき，先の最適パターン発見問題は，すべての可能なパターンを数え上げる素朴なアルゴリズムを用いて，$O(n^{2d+1})$ 時間で解ける．これは，多項式時間であるが，次数が大きく実際の応用には使えない．そこで，入力サイズ n に対して，$O(n)$ から $O(n \log n)$ 時間程度で働く高速なアルゴリズム Split-Merge を開発した [Arimura 00c]．このアルゴリズムは，ほぼランダムなテキストから最適な k 近接 d 語相関パターンを，線形に近い時間で計算するものである．アルゴリズムの詳細は，[安部 00, Arimura 00c] を参照されたい.

[Fujino 00] では，Apriori アルゴリズム [Agrawal 94] のディスク走査のアイディアに基づいて，外部記憶指向の高速な発見アルゴリズム Levelwise-Scan を開発した．このアルゴリズムでは，接尾辞木データ構造とパターン照合オートマトンの技法を用いており，すべての候補パターンを主記憶上において，ディスクにおかれたデータベースを少数回走査することで，効率良いディスク入出力を達成する．さらに，基本属性として許した任意のフレーズからなる膨大な仮説空間を効率良く扱う．また，順序なしと順序ありの両方の近接語相関パターンを扱うことができる.

4·3 Webマイニングへの応用

論文 [安部 00, Arimura 00a] では，最適化パターン発見を，Web データからのキーワード獲得に応用した．未知の語彙をもつ大量のテキストデータに対して，従来のマイニング方式のように，高頻度パターンを抽出するだけでは，データに特徴的なパターンが，自明な高頻度語に隠ぺいされてしまう．情報検索で，伝統的な禁止語リストによる高頻度語の除去も，データマイニングでは有効でない．そこで，マイニング対象の文書集合に加えて，制御群となる文書集合を用いて，最適パターン発見アルゴリズムで，両者の間の情報エントロピーを最小化するパターンを発見する.

実験では，Web 検索エンジンにあいまいな問合せ語を与え，得られたページ集合からさらにリンクをたどって得た大量のページ集合をデータとして用いた．これを，マイニング対象の問合せ語と，対照群を表す問合せ語の両方に行って，データ集合を得て，キーワード発見を行った．これにより，最適化パターン発見は Web マイニングにおいて有効であり，従来の手法における問題点をうまく解決できることがわかった．これにより，ノイズや自明なパターンをキャンセルすることができる．また，単純な一つの単語をキーワードとして用いる方法に対して，任意長のフレーズや，フレーズの連接からなるパターンを扱いながら，計算量的にきわめて高速である点が有利である [安部 00].

5. おわりに

本稿では，Web マイニングを広い意味で，インターネット上のテキストや半構造データから情報を獲得することととらえ，ラッパー抽出や，Web 知識ベースプロジェクト，類似性結合，引用文献データベース，最適化パターン発見など，Web マイニングに関する最近の話題について紹介した.

本稿で概観してきた情報抽出は，ネットワーク上に公

開された一般の情報を対象としており，一度公開された情報の2次利用など，著作権や情報の質に関して，まだまだ難しい点も多い．最近，物理分野では，ロスアラモス国立研究所が中心となり，技術報告・手稿のポストスクリプト・PDFファイルを公開するWebサイト [e-Print 01] を運用しており，このサイトは高エネルギー物理学を中心とする分野で研究成果の迅速な公開・普及に広く利用されているそうである [Taubes 96].

3・3節で紹介したResearchIndexのようなシステムは，公開される無償情報の量が増えると，提供される情報の質も向上する．今後のインターネットの利用形態や，企業活動や研究活動の変化と密接に関連する形で，Webからの情報抽出や情報獲得技術は発展していくと思われる．

謝　辞

九州大学の有川節夫先生，竹田正幸先生，村上義継君，安部潤一郎君，九州工業大学の下薗真一先生には，本稿の内容に関してご議論いただきました．また，東京大学の森下真一先生，日本IBM東京基礎研究所の那須川哲哉氏，北海道大学の山本章博先生にはWebマイニングに関して，貴重な示唆をいただきました．ここに感謝いたします．

◇ 参 考 文 献 ◇

[安部 00] 安部潤一郎, 藤野亮一, 下薗真一, 有村博紀, 有川節夫: テキストデータからの高速データマイニング, 人工知能学会誌, 特集「発見科学」, Vol. 15, No. 4, pp. 618-628 (2000).

[Abiteboul 00] S. Abiteboul, P. Buneman and D. Suciu: Data on the Web: From relations to semistructured data and XML, Morgan Kaufmann, San Francisco, CA (2000).

[Agrawal 94] R. Agrawal and R. Srikant: Fast algorithms for mining association rules, *Proc. VLDB'94,* pp. 487-499 (1994).

[Ambite 00] J. L. Ambite and C. A. Knoblock: Flexible and scalable cost-based query planning in mediators: a transformational approach, *Artificial Intelligence,* Vol. 118, pp. 115-161, (2000).

[Arimura 00a] H. Arimura, J. Abe, R. Fujino, H. Sakamoto, S. Shimozono and S. Arikawa: Text data mining: discovery of important keywords in the cyberspace, *Proc. IEEE Kyoto Int'l Conf. Digital Library, IEEE* (2001). (to appear)

[Arimura 00b] H. Arimura, H. Sakamoto and S. Arikawa: Learning term rewriting systems from entailment, In ILP 2000 Work-in-Progress session (Jul. 2000).

[Arimura 00c] H. Arimura, S. Shimozono and S. Arikawa: Efficient discovery of optimal word-association patterns in large text databases, *New Generation Computing,* Vol. 18, pp. 49-60 (2000).

[Cohen 98] W. W. Cohen: Integration of heterogeneous databases without common domains using queries based on textual similarity, *Proc. ACM SIGMOD Conf. Management of Data* (1998).

[Cohen 00] W. W. Cohen: WHIRL: a word-based information representation language, *Artificial Intelligence,* Vol. 118, pp. 163-196 (2000).

[Craven 00] M. Craven, D. DiPasquo, D. Freitag, A. Mc Callum, T. Mitchell, K. Nigam and S. Slattery: Learning to construct knowledge bases from the World Wide Web, *Artificial Intelligence,* Vol. 118, pp. 69-113 (2000).

[e-Print 01] e-Print archive, the Los Alamos National Laboratory, 2001, http://xxx.lanl.gov/.

[Fujino 00] R. Fujino, H. Arimura and S. Arikawa: Discovering unordered and ordered phrase association patterns for text mining, *Proc. PAKDD2000,* LNAI (2000).

[Fukuda 96] T. Fukuda, Y. Morimoto, S. Morishita, and T. Tokuyama: Data mining using two-dimensional optimized association rules, *Proc. ACM 1996 SIGMOD,* pp. 13-23 (1996).

[Joshi 99] K. P. Joshi, A. Joshi, Y. Yesha and R. Krishnapuram: Warehousing and mining Web logs, *Proc. the 2nd Int'l Workshop on Web information and data management,* pp. 63-68 (1999).

[Kushmerick 00] N. Kushmerick: Wrapper induction: efficiency and expressiveness, *Artificial Intelligence,* Vol. 118, pp. 15-68 (2000).

[Lawrence 99a] S. Lawrence and L. Giles: Accessibility of information on the Web, *Nature,* Vol. 400, pp. 107-109 (1999).

[Lawrence 99b] S. Lawrence and L. Giles: Digital libraries and autonomous citation indexing, *IEEE Computer,* Vol. 32, No. 6, pp. 67-71 (1999).

[Richards 92] B. L. Richards and R. J. Mooney: Learning relations by pathfinding, *Proc. AAAI-92,* pp. 50-55 (1992).

[永田 01] 永田昌明, 平 博順: テキスト分類—学習理論の見本市—, 特集「情報論的学習理論とその応用」, 情報処理, Vol. 42, No. 1, pp. 32-37 (2001).

[那須川 01] 那須川哲哉, 河野浩之, 有村博紀: テキストマイニング基盤技術, 人工知能学会誌, Vol. 16, No. 2, pp. 201-211 (2001).

[Sakamoto 00] H. Sakamoto, H. Arimura and S. Arikawa: Identification of tree translation rules from examples, *Proc. ICGI 2000,* LNAI 1891, pp. 241-255 (Sep. 2000).

[Sakamoto 01] H. Sakamoto, Y. Murakami, H. Arimura and S. Arikawa: Extracting partial structures from HTML documents, *Proc. 14th International FLAIRS Conference,* AAAI press (2001) (To appear).

[Schapire 00] R. Schapire and Y. Singer: BoosTexter: A boosting-based system for text categorization, *Machine Learning,* Vol. 39, No. 2/3, pp. 135-168 (2000).

[Taubes 96] G. Taubes: Electronic preprints point theway to author empowerment, *Science,* Vol. 271, p. 767 (1996).

2001年2月5日　受理

著 者 紹 介

坂本 比呂志（正会員）

1996年3月九州大学大学院システム情報科学研究科情報理学専攻修士課程修了．同年4月日本学術振興会特別研究員（DC1）．1998年12月同研究科博士課程修了．同年1月九州大学大学院システム情報科学研究院情報理学部門助手．現在に至る．機械学習と計算量理論，Web上のテキストデータからの知識獲得の研究に従事．博士（理学）．

有村　博紀（正会員）は，前掲（Vol. 16, No. 2, p. 211）参照．