ブートストラップ法による語の共起を用いたWebからの類似関係抽出

加藤　誠† 大島　裕明† 小山　聡† 田中　克己†

† 京都大学大学院情報学研究科社会情報学専攻 〒 606-8501 京都府京都市左京区吉田本町
E-mail: †{kato,ohshima,oyama,tanaka}@dl.kuis.kyoto-u.ac.jp

あらまし　本稿では，ある関係を満たす語の組をシードとして与えた場合，その関係と類似するような関係を満たす語の組を，自動的に Web から抽出する手法についての提案を行う．我々は，Web 検索エンジンの検索結果として得られるテキストから，与えられた組と同じ関係にある語の組を抽出し，得られた結果を自動的に評価することにより，より精度が高く，多様な抽出手法を生成する．これを繰り返すことで，少ないシードから正確かつ多くの語の組を取得することを目的とする．

キーワード　知識抽出，ブートストラッピング，情報検索

1. はじめに

近年，インターネットと Web 検索エンジンの普及により，様々な情報を Web 文書から取得することが可能となった．これらを利用して，Web に存在する多様で非構造な情報から自然言語処理やデータマイニングなどの技術を用いて，有用な情報を抽出することが盛んに行われている．これは辞書やオントロジーの自動構築などに利用され，特定の語の上位語，下位語，同位語などを取得する手法が数多く提案されている．これらの手法は，入力として 1 語，または，数語を与え，その語と特定の関係にある語を Web から取得することを目的としている．例えば，上位語の取得では，入力として「ペンギン」が与えられれば，出力として「鳥」が得られ，同位語の取得では，入力として「TOYOTA」が与えられれば，出力として「日産」などが得られる．しかし，これらの手法は一般的な関係のみを対象としており，特定の関係にのみに特化している．

そこで本稿では，任意の関係にある語のペアを入力として与え，それと類似した関係にある語のペアを Web から取得する手法を提案する．例えば，(こころ，夏目漱石)，(人間失格，太宰治) などを与えた場合は，タイトルとその著者のペアを，(静岡，茶)，(愛媛，みかん) などを与えた場合は，県名とその特産物のペアを出力する．このような，入力と出力によって，我々は多様な関係を一つの手法で取得することを可能とした．また，少ない入力のペアから類似関係にあるペアを大量に取得するために，我々はブートストラップ法を用いた．これは近年，知識抽出において用いられており，入力として与えられた少数の語を元にして同位語などを大量に取得する手法である．ブートストラップ法では，入力から取得できた語を使って新しい抽出方法を生成することで，取得する語の数を徐々に増加させていくことが可能である．

多くのブートストラップによる知識抽出手法では，言語パターンによる知識抽出が行われているが，本稿では，語の分布の差異を利用した手法を用いる．言語パターンによる抽出は精度が高いが，厳密な一致を求めるために，再現率は低いと考えられている．また，現実世界における多様な関係を考えた際に，言語パターンのみで表現できないものが考えられる．例えば，(京都，八ツ橋) という関係を考えた場合，「八ツ橋は京都の有名なお土産」といったフレーズだけでは表現は難しく，「八ツ橋」は「京都」において，土産である，伝統がある，代表的な菓子である，などといったように，「京都」と「八ツ橋」の間には複雑な関係があることがわかる．これらを考慮した場合，言語パターンによって類似関係抽出を行うことは，条件が厳しく柔軟でないと考えられる．

しかし，ブートストラップに適合率の低い手法を用いれば，得られる結果全体の適合率が大きく低下してしまう．そのため，得られた語のペアが適切であるかどうかを評価し，正解であると確信が得られたもののみを採用し精度を保つ必要がある．本手法では，ある類似関係にあるペア (x, y) の x と y はある特定のクラスに属すという性質，すなわち，得られる全ての $x(y)$ は同位語となるという性質を利用して得られたペアの評価を行う．同位語であるかどうかを判定するために，Web を用いて語の共起度を測る指標の一つである WebPMI を用いた．

2. 関連研究

2.1 特定の関係にある語の抽出

テキストコーパスやデータセットから，特別な関係にある語を自動的に抽出する研究は数多く行われている．Hearst ら [1] は上位語や下位語を発見するために，“such as”のような言語パターンに着目した．Bayesian Sets [2] はベイズ推定を用いて，同位語を共起テーブルのような大規模データから発見するものである．このアルゴリズムは単純かつ高速であるが，EachMovie や Grolier encyclopedia のような大量のデータセットを必要とする．Lin ら [3] は係り受け解析をした大量のテキストデータから類義語を発見する手法について提案している．

特別な関係にある語を発見する研究もいくつか行われている．小山ら [4] はある語の話題について詳細に述べている語を Web から取得した．これらの語の関係は一種の “part-of”関係にあると言える．外間ら [5] は人物の呼称を Web から抽出している．これは，人物の名称の前に現れる言語パターンを用いて候補となる言語表現を発見した後に，それが呼称であるかの評

価を行っている．

このように，特定の関係にある語の抽出について，多くの研究が行われている．我々の手法は，入力として与えられた様々な関係に対して，類似関係にある語を取得することができるため，これらの研究の一般化であるとも言える．

2.2 ブートストラップによる知識抽出

ブートストラップによって特定の関係にある語のペアを取得する研究は既に存在しているが，その多くが言語パターンを用いたものである．Brin [6] は本のタイトルとその著者のペアを少ないシードから大量に取得するために，ブートストラップを用いている．これは Web 文書中にシードが「prefix，著者名，middle，タイトル名，suffix」というパターンで出現している場合，パターンに適合する語列が同様の関係にあるペアであると仮定し，これを繰り返すことで目的とする語のペアを取得している．Snowball [7] は Brin と同様の処理を行うシステムで，Brin の prefix，suffix などのパターンに対して重みをつけ精度の向上を図っている．張ら [8] はレコード抽出を行う文書をユーザの意図に適合した文書に限定することによって，レコード抽出における二つの問題を解決している．1 つ目の問題は，テキスト処理にかかるコストが大きいことである．多くの文書からレコード抽出を行う場合，この問題は顕著となる．張らはユーザの意図に適合した文書を優先的に処理することによって，抽出効率を向上することに成功している．もう 1 つの問題は，得られた結果のすべてがユーザの意図に合致しないことである．これに対して，得られたレコードの正否判定のみでなく，ユーザの意図に合致しているかどうかも踏まえて評価することによって，ユーザの要求により適合したレコードを抽出している．

KnowItAll [9], [10] は “such as ”や “and other”などの言語パターンを用いて，入力されたシードと同じクラスに属する語，すなわち，同位語をブートストラップを用いて取得するシステムである．KnowItAll では，ブートストラップを用いて適合，不適合な語を大量に取得し，候補語がどのようなパターン内で現れる場合には適合であるかを学習している．Riloff ら [11] はブートストラップを同位語の抽出とパターン発見にそれぞれ用いる Multi-level ブートストラップを提案している．

最近では，言語パターンのみではなく HTML データ構造を利用してブートストラップによる知識抽出が行われている [12] [13]．これらは，Web 文書中の表や箇条書きなどの HTML データ構造にも着目し，パターンに HTML タグを含めることによって，知識抽出を行っている．

2.3 関係の類似

Turney ら [14] は語 A と語 B の組が与えられたときに，予め用意された複数の語 C と語 D の組のうち，A と B の関係と最も近い C と D の組を選択するという比例アナロジーの問題に対して様々なアプローチによる研究を行ってきた．その中でも最も効果的であることが示されたのは，ベクトル空間モデルを用いた手法である．この手法は，予め用意された複数の言語パターン，例えば，“A of B”や “B to A”などに対応した次元を持つベクトルを用意し，その要素をそれらの言語パターンを満たす文書数としたもので A と B の組を表現する．同様にして，用意された C と D の各組をベクトルによって表現し，これと A，B の組を表すベクトルとのコサイン類似度を取る．最後に，ベクトル間のコサイン類似度を関係の類似度と見なし，最も類似度が高い C，D の組を正解とするのが，ベクトル空間モデルを利用して提案された手法である．Bollegala ら [15] は，LRA では 8 日以上かかっていた同問題 374 問を，Web からの言語パターン抽出と SVM による学習で，6 時間以下に短縮し高速化を行った．これらの研究は，すでに与えられた解候補の中から最も適したものを選択する問題に焦点を当てており，Web から類似関係を抽出するという我々の目的とは異なるものである．一方，入力として語 A，B，C を与え，A と B の関係が C と D の関係と類似するような語 D を発見する研究が大島ら [16] によって始められている．これは語 A，B 間の言語パターンを抽出し，その言語パターンに対して語 C を適用することによって，高速に語 D に該当する語を発見することを目的としている．

3. 語の共起を用いた Web からの類似関係抽出

類似関係を議論するために，関係についていくつかの定義を示す．n 項関係は n 個の定義域 $X_1, X_2, \ldots, X_n$ に対して，その直積の部分集合で表現される．すなわち，任意の n 項関係 R は $R \subset X_1 \times X_2 \times \ldots \times X_n$ となる．我々は，ある要素 $(x_1, x_2, \ldots, x_n)$ が与えられたときに，これらで成り立つような関係の集合を以下のように表す．

$$\begin{aligned}&\mathrm{Relation}(x_1, x_2, \ldots, x_n)\\ &= \{R | R \subset X_1 \times X_2 \times \ldots \times X_n \wedge R(x_1, x_2, \ldots, x_n)\} \quad (1)\end{aligned}$$

例えば，「日本」と「東京」の間には様々な関係が存在する．首都である，経済の中心である，政治の中心である，都道府県の一つである，などといった関係がそれらである．これら一つ一つの関係は R に対応し，これらの集合が Relation(日本，東京) に相当する．R には様々な粒度が存在し，例えば，「日本」と「東京」に成り立つ関係を，都道府県の一つという関係とするのか，part-of 関係とみなすのか，といった問題がある．

我々は，入力として語 A，B，C を与え，A と B の関係に類似するような C に対する語 D を Web から発見する研究を行ってきた [17]．語 D は語 C との組み合わせの内，語 A，B の関係と類似度が高い順にソートされ出力される．入力 $q = (A, B, C)$ と，語 d_i に対するランク関数 Rank は以下のようにして表される．

$$\begin{aligned}&\mathrm{Rank}(q, d_i)\\ &= \mathrm{Sim}(\mathrm{Relation}(A, B), \mathrm{Relation}(C, d_i)). \quad (2)\end{aligned}$$

A と B，C と D の関係の類似は，それらの間で成り立つ関係の集合を考慮しなくてはならない．例えば，日本と東京の間に成り立つ関係は多数存在するため，ある特定の関係，例えば首都の関係だけを考慮した類似度にはならない．ただし，我々の実装では Web の語分布を用いているため，より多く言及されている関係に重みが高くなるので，全ての関係を平等に扱っているわけではない．

実装では，語 A と B を強く結びつけるような語 t を Web から発見し，語 t と C が出現するときにのみ有意に多く出現するような語が，条件を満たすような語 D である可能性が高いとし，入力 A,B と類似した関係を Web から取得している．以下，語 A，B を強く結びつける語を A，B の関係接続語と表現する．例として，入力として語 A = 静岡，B = お茶，C = 愛媛が与えられた場合を考える.「静岡」と「お茶」の関係接続語としては,「産地」「直送」などといった語が抽出できるが，関係接続語は「静岡」と「お茶」の間に成り立つ関係そのものを表現しているわけではない.「静岡」と「お茶」の間で成り立つ関係の一つに，後者が前者の特産物である，という関係が存在しているために,「静岡」と「お茶」が出現するときにのみ「産地」や「直送」などといった語が多く現れるだけである．A と B の関係接続語は A と B で成り立つ関係と意味的に一致しているとは限らない．関係接続語集合 T を発見した後,「静岡」,「お茶」と関係が類似するような「愛媛」に対する語 D を発見する．最終的に，T の要素 t と「愛媛」が出現するときのみ出現頻度が高くなるような語として「みかん」などが得られる．本稿では，この手法とブートストラップ，WebPMI による評価により Web から大量に類似関係を取得する．

入力 A，B，C に対して語 D を発見する手法の概要は以下の通りである．

3.1 入力 A と B の関係接続語の発見

（1） 入力 A 及び入力 B に対して，A を含み B を含まない文書を検索するクエリと，B を含み A を含まない文書を検索するクエリで Web 検索を行い，検索結果のタイトルとスニペットを取得する．

（2） 入力 A と入力 B を含む文書を検索するクエリで Web 検索を行い，検索結果のタイトルとスニペットを取得する．

（3） タイトルとスニペットに対して形態素解析を行い，形態素ごとに分割する．また，ストップワードリストを用いて不要な語を除去する．

（4） 品詞が「名詞」であると推定された各語 t_i に対して，“語 A を含み語 B を含まない文書と語 A と B を共に含む文書において，語 t_i の出現確率が等しい”という帰無仮説に対して χ^2 検定を行う．同様に，語 B を含み語 A を含まない文書と語 A と B を共に含む文書に対しても検定を行う．

（5） 有意水準 α の検定において棄却された語 t_i のうち，語 A，B が出現したときに出現確率が高くなるものを，入力された語 A と B の関係接続語として採用し，これを関係接続語集合 T とする．

(1)〜(5) までの手法は，語 A と B が現れたときのみに有意に多く出現するような関係接続語集合 T を Web から発見するものであり，以下，入力として語 A と B，パラメータ α が与えられたとき，関係接続語集合 T を出力するプロセスを $\mathrm{RelationalConnectingTerm}_{\alpha}(A,B)$ ($\mathrm{RCT}_{\alpha}(A,B)$) と表記する．

3.2 入力 C に対する語 D の発見

（6） 関係接続語集合 T の全ての語 t_i に対して，入力 C を含み語 t_i を含まない文書を検索するクエリと，t_i を含み C を含まない文書を検索するクエリで Web 検索を行い，検索結果のタイトルとスニペットを取得する．

（7） 関係接続語集合 T の全ての語 t_i に対して，入力 C と語 t_i を含む文書を検索するクエリで Web 検索を行い，検索結果のタイトルとスニペットを取得する．

（8） タイトルとスニペットに対して形態素解析を行い，形態素ごとに分割する．また，ストップワードリストを用いて不必要な語を除去する．

（9） 品詞が「名詞」であると推定された各語 d_j に対して，“語 C を含み語 t_i を含まない文書と語 C と t_i を共に含む文書において，語 d_j の出現確率が等しい”という帰無仮説に対して χ^2 検定を行う．同様に，語 t_i を含み語 C を含まない文書と語 C と t_i を共に含む文書に対しても検定を行う．両者の検定の結果，帰無仮説が発生する確率をそれぞれ $P_C(d_j)$，$P_{t_i}(d_j)$ とする．

（10） 有意水準 β の検定において棄却された語 d_j のうち，語 C，t_i が出現したときに出現確率が高くなるものに対して，$P_C(d_j)$，$P_{t_i}(d_j)$ の積を $P_{C,t_i}(d_j)$ とする．

（11） 関係接続語集合 T の全ての語 t_i に対する，$P_{C,t_i}(d_j)$ の全ての積を語 d_j のスコア $\mathrm{Score}_C(d_j)$ とする．

(6)〜(11) までの手法は関係接続語集合 T を利用した語 C に対する語 D の発見であり，以下，入力として関係接続語集合 T と語 C，パラメータ β が与えられたとき，順序付き語集合の上位 k 件を出力するプロセスを $\mathrm{SimilarRelationTerm}_{\beta,k}(C,T)$ ($\mathrm{SRT}_{\beta,k}(C,T)$) と表記する．

この手法は，語集合 T によって A と B が共起するような文脈を表現し，その文脈下において語 C とよく共起する語が A と B の関係と類似するような，C に対する D であるという仮定を利用している．語の出現分布を用いることで，言語パターンよりも複雑な関係に対して適応しやすいと考えている．

4. ブートストラップ法による Web からの類似関係抽出

4.1 プロセス概要

前節で用いたプロセス RelationalConnectingTerm，SimilarRelationTerm を用いて，少数のシードから類似関係を抽出する手法を提案する．本稿では前ステップで得られた出力を入力として与え，再帰的操作によって徐々に解を増やしていく手法であるブートストラップ法を用いる．ブートストラップ法を用いる際には得られる出力の精度を高く保つことが必要である．前のステップで得られた出力にノイズが含まれている場合，次のプロセスではノイズが入力として与えられるため，連鎖的に精度が低下してしまう危険性がある．そのために，解の選択は慎重に行い，抽出手法に関しても精度のよいものだけを利用することが望ましい．そこで，本稿では WebPMI により得られた解の候補及び関係接続語集合を評価し，確実に正解であると判断できる解のみにを次回のプロセスにおける入力として与える．

本稿で行うブートストラップの概要は擬似コードで書かれたアルゴリズム 1 の通りである．また，図 1 に $Seeds$ = {(静岡

アルゴリズム 1 Web からの類似関係抽出

```
1: // Definition
2: RCT_XX: Relational Connecting Term between X - X
3: RCT_XY: Relational Connecting Term between X - Y
4: RCT_α(x, y): RelationalConnectingTerm_α(x, y)
5: SRT_β,k(x, T): SimilarRelationTerm_β,k(x, T)
6:
7: // 1. Input seeds
8: Pairs ← Seeds
9:
10: while Pairs are not enough do
11:
12:    // 2. Extract connecting terms to find coordinate terms
13:    for all p_1, p_2 in Pairs do
14:       RCT_XX ← RCT_XX ∪ RCT_α(p_1[X], p_2[X])
15:    end for
16:
17:    // 3. Find coordinate terms with connecting terms
18:    for all p in Pairs, T in PowerSet(RCT_XX) do
19:       SetXCands ← SetXCands ∪ SRT_β,k(p[X], T)
20:    end for
21:
22:    // 4. Choice the good coordinate terms
23:    for all x in SetXCands do
24:       if SRTE_γ1(x, RCT_XX, Pairs[X]) > δ_1 then
25:          SetX ← SetX ∪ {x}
26:       end if
27:    end for
28:
29:    // 5. Extract connecting terms to find similar relation terms
30:    for all p in Pairs do
31:       RCT_XY ← RCT_XY ∪ RCT_α(p[X], p[Y])
32:    end for
33:
34:    // 6. Find similar relation pair with connecting terms
35:    for all x in SetX, T in PowerSet(RCT_XY) do
36:       PairCands ← PairCands ∪ {(x, y)|y ∈ SRT_β,k(x, T))
37:    end for
38:
39:    // 7. Choice the good similar relation pairs
40:    for all p in PairCands do
41:       if SRTE_γ2(p[Y], RCT_XY, SetX) > δ_2 then
42:          Pairs ← Pairs ∪ {p}
43:       end if
44:    end for
45:
46: end while
```

，お茶), (愛媛，みかん)} を入力したときの動作例を示す．図 1 の 1 から 7 までの番号はアルゴリズム 1 のコメント番号にに対応している．

最初に，$Seeds$ を入力として与える．$Seeds$ の定義は $Seeds \subset X \times Y$ となる．すなわち，$Seeds$ は集合 X と集合 Y の要素のペアになる．例えば，$Seeds$ = {(静岡，お茶), (愛媛，みかん)} であれば，集合 X は県名であり，集合 Y が

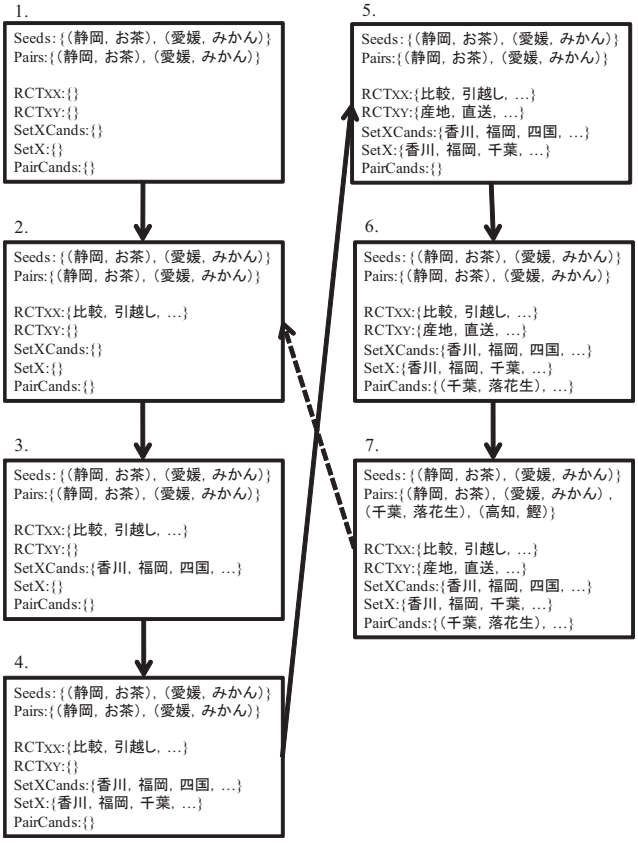

図 1　Web からの類似関係抽出の例

特産物であることが暗に与えられる．これは，$Pairs$ の定義とも等しく，$p = (x, y) \in Pairs$ のとき $p[X] = x$，$p[Y] = y$，また，$Pairs[X] = \{p[X]|p \in Pairs\}$，$Pairs[Y] = \{p[Y]|p \in Pairs\}$ と定義する．初期操作として，与えられた $Seeds$ は正解と推定されたペア集合 $Pairs$ に追加される．以上の操作がアルゴリズム 1 の (1. Input seeds) に当たる．

$Pairs$ が十分な量に達していなければ，以下のプロセス (2-7) を繰り返し行う．ただし，これ以降 $Pairs$ の要素のうちプロセスで用いるのは評価値の高い上位 n 件のみに限定する．

まず，集合 X に含まれると予想される要素を，語の共起を用いた Web からの類似関係抽出手法によって取得する．$Pairs$ = {(静岡，お茶), (愛媛，みかん)} の例の場合，A = 静岡，B = 愛媛，C = 静岡という入力を与えることによって，集合 X に含まれるような要素，すなわち，静岡や愛媛の同位語を取得することが出来る．ブートストラップによる類似関係抽出では，あらかじめ関係接続語集合 RCT_{XX} を $Pair$ に含まれる X の全ての組み合わせから求め，RCT_{XX} の部分集合から得られる上位 k 件の X 要素を求め，これを $SetXCands$ とする．$SetXCands$ の各要素を WebPMI によって評価し，有効な関係接続語集合及び新しい X 要素集合 $SetX$ を決定する．以上の操作がアルゴリズム 1 の 2. から 4. に当たる．

次に，得られた集合 $SetX$ の要素から，シードと同じような関係にある集合 Y の要素を発見する．$Pairs$ = {(静岡，お茶), (愛媛，みかん)}，$SetX$ = 香川，福岡，千葉, ... の例の場合，A = 静岡，B = お茶，C = 香川という入力を与えること

によって，香川に対する集合 Y の要素を発見する．$Pair$ に含まれる全ての組から得られた関係接続語集合 RCT_{XY} の部分集合から得られる上位 k 件の Y 要素を求め，これと X 要素の組集合を $PairCands$ とする．$PairCands$ の各要素を WebPMI によって評価し，有効に働く関係接続語集合と $Seeds$ と類似した関係を持つ組集合 $Pairs$ を得る．以上の操作がアルゴリズム 1 の 5. から 7. に当たる．

以上がブートストラップによる類似関係発見手法の概要である．

4.2 抽出手法及びペア候補の評価

ブートストラップにより多様な抽出手法を生成してその精度を保つためには，抽出手法の評価が必要となる．本手法における抽出手法とはすなわち，新たに語を発見するのに用いる関係接続語の種類を指している．また，語の共起を用いた Web からの類似関係抽出手法は精度の面で不十分であるため，ブートストラップ法で多くの類似関係を抽出するためには，正解と判断できたものだけを選択する必要がある．そこで，類似関係を抽出した後にペアの候補を評価し，以下の 2 つの仮定に基づいて抽出手法及びペア候補の評価の評価値を求める．

- 良いペアを得られた抽出手法は良い抽出手法である．
- 良い抽出手法から得られたペアは良いペアである．

本稿では，ある類似関係にあるペア (x, y) の x と y はある特定のクラスに属すという性質，すなわち，得られる全ての $x(y)$ は同位語となるという性質を利用して，得られたペアを評価したあとに良い抽出手法を評価する．最終的には，良い抽出手法から得られた良いペアが解として選択される．

ある同位語集合 T に対して語 t' がその同位語であるかどうかを判定するために，我々は Web を用いて語の共起度を測る指標の一つである WebPMI を用いた．WebPMI は二語の意味的類似度を判定するために Bollegala ら [18] によって用いられており，その定義は以下のようになる．

$$\mathrm{WebPMI}(P,Q) = \begin{cases} 0 & \text{if } \mathrm{H}(P \cap Q) \leqq c \\ \log_2 \left(\frac{\frac{\mathrm{H}(P \cap Q)}{N}}{\frac{\mathrm{H}(P)}{N} \frac{\mathrm{H}(Q)}{N}} \right) & \text{otherwise.} \end{cases} \quad (3)$$

ただし，$\mathrm{H}(x)$ はクエリ x で検索を行ったときのヒット件数であり，N は検索エンジンの全インデックスページ数である．

語 t' が語集合 T と同位関係にあるかを判定するために，この WebPMI をある語 t' と語集合 T の各要素間で求めたときの平均を利用した．提案手法の中ではこれを利用し，シードから得られる語集合 $Seeds[Y] = \{p[Y] | p \in Seeds\}$ と全ての要素とある語 t' の共起度を WebPMI によって計算し，その共起度の平均を集合 $Seeds[Y]$ の要素間の平均で正規化した値がある程度大きければ，語 t' は集合 $Seeds[Y]$ と同じクラスに属する，すなわち，同位語であると判定した．

ある同位語集合 T に対して語 t' がその同位語であるかどうかを，WebPMI を用いて評価する関数 $\mathrm{WE}(t', T)$ を以下のように定義する．

$$\mathrm{WE}(t', T) = \frac{1}{|T| \mathrm{InnerWE}(T)} \sum_{t \in T} \mathrm{WebPMI}(t, t') \quad (4)$$

ただし，正規化を行うための同位語集合内での WebPMI 平均，$\mathrm{InnerWE}(T)$ は以下の通りである．

$$\mathrm{InnerWE}(T) = \frac{1}{m} \sum_{\substack{t_1, t_2 \in T \\ (t_1 \neq t_2)}} \mathrm{WebPMI}(t_1, t_2) \quad (5)$$

式中の m は同位語集合の任意の 2 語の組み合わせ数であり，$m = {}_{|T|}C_2$ となる．

以上で定義された指標を用いて抽出手法及びペア候補の評価関数を定義する．抽出手法，すなわち，関係接続語集合 T が語集合 X に対して語集合 $SetY = \bigcup_{x \in X} \mathrm{SRT}_{\beta,\mathrm{k}}(x, T)$ を得たとき，語集合 $SetY$ は $Seeds[Y]$ (もしくは $Seeds[X]$) の同位語でなくてはならない．そこで抽出手法の評価関数 $\mathrm{RCTE}(T, X)$ には各 $y \in SetY$ と $Seeds[Y]$ の同位語評価値の平均を採用する．

$$\mathrm{RCTE}(T, X) = \frac{1}{|Y|} \sum_{y \in SetY} \mathrm{WebPMI}(y, Seeds[Y]) \quad (6)$$

良いと判断された抽出手法の評価に基づいてペアの評価は行われる．そのため，ある閾値 γ 以上の評価値を得た抽出手法から得られたペアにのみ評価値を与える．与えられたペア $p = (x, y)$ の評価関数 SRTE_γ(y, T, X) は以下のように定義される．

$$\mathrm{SRTE}_\gamma(y, T, X) = \begin{cases} \mathrm{WE}(y, Seeds[Y]) & (\mathrm{RCTE}(T, X) > \gamma) \\ 0 & (\text{otherwise}) \end{cases} \quad (7)$$

評価関数 SRTE がある閾値 δ 以上ならば解として妥当であると判断し採用する．精度良く抽出するためには手法中で用いられる γ_1，γ_2，δ_1，δ_2 を適切に設定する必要がある．

5. 実　　験

ブートストラップによる類似関係抽出手法の精度を確認するため，我々は県名と特産物のペアを与え，その関係と類似したペアを抽出する実験を行った．使用したシードは {(静岡，お茶), (愛媛，みかん)} の 2 組であり，類似関係検索に利用されるパラメータ α 及び β は事前実験より両方共に 0.01 に設定した．本実験の検索で取得する Web 文書数は 100 件であり，類似関係検索には得られた結果の上位 5 件を採用した ($k = 5$)．ブートストラップで用いられる評価値の閾値 γ_1，γ_2，δ_1，δ_2 は，反復 1 回の事前試行によりそれぞれ 0.9，0.9，0.9，1.2 に設定した．また，反復試行に用いる $Pairs$ の要素は評価値の高い上位 $n(= 5)$ 件のみに限定する．

ブートストラップによる反復を 5 回行ったときの結果に対して人手で評価を行い，5 回目に得られた一意な解を全正解集合と仮定したときの適合率-再現率グラフを図 2 に示す．また，反復回数と得られた総正解数の増加の様子を図 3 に示す．

1 回目の反復試行では適合率を 90%弱に保ち，再現率を初期シードの数十倍に増加させている．2 回目の反復試行では適合率を 65%程度まで低下させている一方，正解ペアを多く獲得す

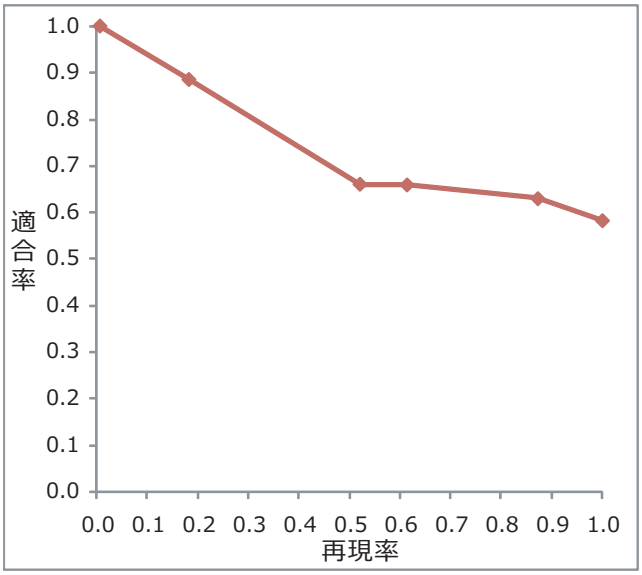

図 2　ブートストラップによる類似関係抽出の適合率-再現率グラフ

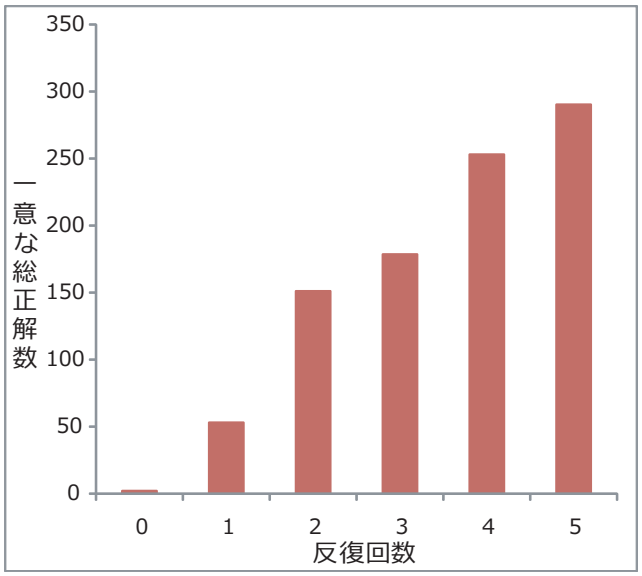

図 3　ブートストラップの反復回数と一意な総正解数の増加

ることに成功している．3 回目，5 回目の試行では正解数を大きく増大させることはないものの適合率は 60%台を保ち，4 回目の試行においても多くの正解を得ていることがわかる．

2 回目の試行以降適合率を損なう要因はいくつか考えられる．第一に，1 回目の反復にはシードが用いられているが，2 回目以降は新たに得られた解からシードを生成するため，最初に入力したシードとは徐々にその関係が異なってくるためであると考えられる．しかし，シードを変えること行わなければ，同じような解が多く得られ再現率の向上につながらない．そのため，反復試行のシードに初期シードを用いるか新しいシードを用いるかは，適合率と再現率のトレードオフの関係に対応している．第二に，1 回目の試行で適切であったパラメータが 2 回目以降は適切でなくなる可能性がある．本手法ではパラメータが多く用いられるが，これらを正確に決定することは困難である．

また，2 回目の試行において大きく適合率を低下させつつも，

表 1　ブートストラップにより得られた正解例

X(県名)	Y(特産物)	X(県名)	Y(特産物)
愛媛	伊予柑	愛媛	温州みかん
岡山	桃	岡山	ピオーネ
沖縄県	ドラゴンフルーツ	沖縄県	パッションフルーツ
宮崎	マンゴー	岐阜	富有柿
京都府	宇治茶	宮崎	パパイヤ
熊本	馬刺し	宮崎県	すいか
広島	牡蠣	京都府	京野菜
香川県	ポンカン	熊本	晩白柚
高知	かつお	熊本	塩トマト
佐賀	ほのか	広島県	レモン
三重	松阪牛	高知	土佐文旦
山口	下関うに	高知県	生姜
山梨県	ぶどう	三重	伊勢茶
滋賀県	赤こんにゃく	山口	車えび
鹿児島	黒豚	山口	長門ゆず
静岡県	マスクメロン	滋賀	鮒寿司
大分	椎茸	鹿児島	さつま揚げ
長崎県	びわ	鹿児島	黒酢
長野	リンゴ	大分県	カボス
鳥取	らっきょう	長崎	ザボン
徳島	鳴門金時	鳥取	二十世紀梨
奈良	富有柿	島根県	いちじく
福岡県	明太子	徳島	すだち
兵庫県	いちご	福岡県	苺
和歌山	南高梅	和歌山	有田みかん

3 回目においてその適合率を下げなかったのは，反復試行に用いる $Pairs$ の数を評価値の高い上位 $n(=5)$ 件に限定しているため，信頼度の低い解が反復プロセスに影響しなかったためだと考えられる．

表 1 にブートストラップによる類似関係発見で得られた正解の一部を示す．不適合であるペアの多くは Y (特産物) の誤りによるもので，都道府県名の誤りは数例しか見られなかった．また，シードとして {(静岡，お茶),(愛媛，みかん)} を指定したために，中部以西の地域のみしか都道府県名を得ることができなかった．これは，2 個のペアを与えるだけでは正確には同位語の粒度，すなわち，どの上位語の下位語に当たる同位語であるかということを暗に与えることができなかったためであると考えられる．シードとして，(青森，リンゴ) などを加えれば全ての都道府県名が得られると予想される．

また，シード数が 2 個のみであったために「みかん」と WebPMI が高いような柑橘系の特産物が多く得られるという結果になった．これも同様にシード数を適当な数に増やすことによって解決できる問題であると考えられる．

表 2 に各反復試行で用いた関係接続語集合 (RCT_{XY}) の評価値が高い上位 3 件を示す．1 回目から 3 回目の試行では特産物関係を結びつけるような語集合が得られているが，2 回目，3 回目からは果物に限定されるようになり，4 回目，5 回目の試行では柑橘系の固有名詞が出現しているため，これがノイズと

表 2　各反復回の関係接続語集合 (上位 3 件)

	関係接続語集合	評価値 (RCTE)
1	直送	1.17
	産地 直送	1.11
	ギフト 直送	1.09
2	果物	1.24
	特産品	1.17
	果物 特産品	1.15
3	果物	1.26
	果物 産地	1.21
	産地	1.20
4	果物 産地 清見オレンジ	1.25
	果物 清見オレンジ	1.25
	果物	1.25
5	雲仙レモネード 果物 清見オレンジ	1.24
	果物	1.21
	レモン果汁 雲仙レモネード 果物	1.21

なって精度の悪化に関与していると考えられる．

6. ま と め

本稿では，任意の関係にある語のペアを入力として与え，それと類似した関係にある語のペアを Web から取得する手法を提案した．類似関係の抽出には語の分布の差異を利用した手法を用い，精度の低下を防ぐために抽出されたペア及び抽出手法を WebPMI を用いて評価した．都道府県名と特産物の関係にあるシードを入力として与えた実験において，1 回目の試行では高い適合率での類似関係抽出に成功し，2 回目以降の反復では大きく適合率を損なうことなく大量の関係を取得することに成功した．

しかし，我々が扱った問題は様々な要因が存在するために手法やパラメータが複雑化し，実際に手法を用いる際にはパラメータの調整が必要となる．そのため，どのような入力に対してもパラメータを調整することなく類似関係を抽出できるような，頑健なブートストラップ手法が必要であると考えている．

謝辞

本研究の一部は，京都大学グローバル COE プログラム「知識循環社会のための情報学教育研究拠点」，および，文部科学省科学研究費補助金特定領域研究「情報爆発時代に向けた新しい IT 基盤技術の研究」，計画研究「情報爆発時代に対応するコンテンツ融合と操作環境融合に関する研究」（研究代表者：田中克己，A01-00-02，課題番号 18049041），および，文部科学省研究委託事業「知的資産の電子的な保存・活用を支援するソフトウェア技術基盤の構築」，異メディア・アーカイブの横断的検索・統合ソフトウェア開発（研究代表者：田中克己）によるものです．ここに記して謝意を表します．

文　献

[1] M. Hearst: “Automatic Acquisition of Hyponyms om Large Text Corpora”, Proceedings of the 14th International Conference on Computational Linguistics (COLING 1992), pp. 539–545 (1992).

[2] Z. Ghahramani and K. A. Heller: “Bayesian Sets”, Proceedings of the 19th Annual Conference on Neural Information Processing Systems (NIPS 2005), pp. 435–442 (2005).

[3] D. Lin: “Automatic Retrieval and Clustering of Similar Words”, Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics (ACL 1998), pp. 768–774 (1998).

[4] S. Oyama and K. Tanaka: “Query Modification by Discovering Topics from Web Page Structures”, Proceedings of the 6th Asia-Pacific Web Conference (APWeb 2004), pp. 553–564 (2004).

[5] T. Hokama and H. Kitagawa: “Extracting Mnemonic Names of People from the Web”, Proceedings of the 9th International Conference on Asian Digital Libraries (ICADL 2006), pp. 121–130 (2006).

[6] S. Brin: “Extracting Patterns and Relations from the World Wide Web”, The World Wide Web and Databases, International Workshop (WebDB’98), pp. 172–183 (1998).

[7] E. Agichtein and L. Gravano: “*Snowball*: Extracting Relations from Large Plain-Text Collections”, Proceedings of the Fifth ACM Conference on Digital Libraries, pp. 85–94 (2000).

[8] 張建偉, 石川佳治, 北川博之：“トピックを考慮した大規模文書情報源からのレコード抽出”, 情報処理学会論文誌 (2007).

[9] O. Etzioni, M. J. Cafarella, D. Downey, S. Kok, A.-M. Popescu, T. Shaked, S. Soderland, D. S. Weld and A. Yates: “Web-Scale Information Extraction in KnowItAll: (Preliminary Results)”, Proceedings of the 13th international conference on World Wide Web, pp. 100–110 (2004).

[10] S. Soderland, O. Etzioni, T. Shaked and D. Weld: “The Use of Web-based Statistics to Validate Information Extraction”, Proceedings of AAAI 2004 Workshop on Adaptive Text Extraction and Mining (ATEM’04) (2004).

[11] E. Riloff and R. Jones: “Learning Dictionaries for Information Extraction by Multi-Level Bootstrapping”, Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI-99), pp. 474–479 (1999).

[12] 水口弘紀, 河合英紀, 土田正明, 久寿居大：“Web 知識を利用したブートストラップによる辞書増殖手法”, 電子情報通信学会第 18 回データ工学ワークショップ論文集 (DEWS2007).

[13] 楠村幸貴, 土方嘉徳, 西田正吾：“テンプレートの交叉と DOM 構造の解析による情報抽出手法の提案”, 電子情報通信学会第 17 回データ工学ワークショップ論文集 (DEWS2006) (2006).

[14] P. D. Turney and M. L. Littman: “Corpus-based learning of analogies and semantic relations”, Machine Learning, **60**, 1-3, pp. 251–278 (2005).

[15] D. Bollegala, Y. Matsuo and M. Ishizuka: “WWW sits the SAT-Measuring Relational Similarity on the Web”, ECAI, pp. 333–337 (2008).

[16] 大島裕明, 田中克己：“両方向構文パターンを用いた Web 検索エンジンからの高速関連語発見手法”, 情報処理学会研究報告, **2008**, 88, pp. 37–42 (2008).

[17] 加藤誠, 大島裕明, 小山聡, 田中克己：“語の共起を用いた Web の類似関係検索”, Web とデータベースに関するフォーラム (WebDB Forum 2008) (2008).

[18] D. Bollegala, Y. Matsuo and M. Ishizuka: “Measuring Semantic Similarity between Words Using Web Search Engines”, Proceedings of the 16th International World Wide Web Conference (WWW 2007), pp. 757–766 (2007).