社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2004－ＦＩ－ 76 (5)
2004－ＮＬ－163 (5)
2004／9／16

HTML文書中の箇条書きとその表題に注目した下位語の自動獲得

新里 圭司　鳥澤 健太郎
北陸先端科学技術大学院大学 情報科学研究科
{skeiji, torisawa}@jaist.ac.jp

本稿では，Web上にあるHTML文書から予め指定された単語もしくは複合語について，その下位語を獲得する方法について述べる．本研究では，「HTML文書中に現れる箇条書きや表の表題には，それらの要素に共通の上位語が含まれやすい」という仮説を設け上位下位関係の獲得を試みる．本手法は，我々が以前に提案した，箇条書きや表の要素に共通する上位語を，統計量を用いて求める手法を拡張することで，予め指定された単語や複合語を上位語として持つ下位語の獲得を可能にする．実験により，従来手法であるパターンベースの手法も含め，他の手法では獲得できない多数の上位下位関係を本手法では獲得できることを示す．

Automatic acquisition of hyponyms based on headings and itemizations in HTML documents

Keiji Shinzato　Kentaro Torisawa
School of Information Science, Japan Advanced Institute of Science and Technology
{skeiji, torisawa}@jaist.ac.jp

This paper describes a method to acquire hyponyms for given hypernyms from HTML documents on the Web. We assume that a heading of an itemization in an HTML document is likely to contain a hypernym of the items in the itemization, and we try to acquire hyponymy relations based on this assumption. Our method is obtained by extending our proposed method where a common hypernym for items in itemizations in HTML documents is obtained by using statistical measures. We empirically show that our new method can obtain a significant number of hyponymy relations which would otherwise be missed by alternative methods.

1　はじめに

近年，多種多様な自然言語処理技術が利用されるようになってきたが，より知的で高度な処理を行うためには，単語間の上位下位関係（*hyponymy relation*）に代表されるような知識がまだまだ不足しており，その獲得は今後ますます重要なものになるといえる．そこで，本稿ではWeb上のHTML文書から単語間の上位下位関係を自動獲得する手法について述べる．Millerら [4] によれば，単語Aが単語Bの上位語（*hypernym*）である（または，単語Bが単語Aの下位語（*hyponym*）である）とは，*"B is a (kind of) A"* がいえる時であると定義されており，本研究でもこの定義に従う．

従来より単語間の上位下位関係の自動獲得について多くの研究がなされてきたが，そのほとんどは *"NP such as NP"* に代表される構文パターン（*lexico-syntactic pattern*）を利用するものであった [3, 1, 8, 2, 5, 7]．しかしこれらの手法は，構文パターンがコーパス中に比較的現れ難いため，大量の上位下位関係を獲得することが難しいという問題を抱えていた．そこで我々は，この問題を解決するために，従来手法とは全く異なるアプローチによる獲得手法として，HTML文書中に現れる箇条書きと，文書頻度などの統計量を用いて上位下位関係を獲得する手法を提案した [6]．実験を通して，構文パターンを用いた手法では獲得できないような上位下位関係を，提案手法により獲得できることは確認されたが，同一の箇条書きに対して複数の上位語を求めることができないという問題や，ユーザの希望する上位下位関係を獲得することが難しいという問題があった．そこで本稿では，我々が以前に提案したこの手法を拡張し，これらの問題点を解決する手法について述べる．以下，我々が以前に提案した手

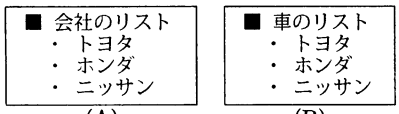

図 1　HTML 文書中に現れる箇条書きの例

法を **AHRAI**（Algorithm for Hyponymy Relation Acquisition from Itemizations），本稿で新しく提案する手法を **HEAIH**（Hyponym Extraction Algorithm from Itemizations and Headings）と呼ぶ．

両手法の決定的な違いは，AHRAI が箇条書きに付与されている表題を無視するのに対し，HEAIH は上位下位関係獲得の手がかりとして表題を利用する点である．この差異は上位下位関係の獲得において大変重要であり，これにより HEAIH ではある下位語が複数の上位語を持つということを同時に認識しやすくなっている．以下，図 1 に示した箇条書きを例に，HEIAH と AHRAI の違いについて述べる．一般に，下位語は複数の上位語を持つと考えられる．例えば，表現「トヨタ」は，少なくとも「車」と「会社」といった 2 つの上位語をとることが可能である．AHRAI は箇条書き中の各表現に共通する上位語を，箇条書き 1 つに対して 1 つだけ出力する．そのため，図 1(A) に示した箇条書きから，「トヨタ」，「ホンダ」，「ニッサン」の上位語として，例えば「会社」を獲得することができるかもしれない．しかし，それと同時に「車」もまた各表現の上位語であるということは認識できない．また，図 1(B) に示す箇条書きが与えられた場合，表題により箇条書きの要素が「車」であると示されているにも関わらず，AHRAI は各要素の上位語として「会社」を獲得することしかできない．これは，AHRAI が箇条書きに付与された表題を上位語獲得の手がかりとして利用しないため及び，図 1(A)，(B) に示した箇条書きの要素が同一であるためである．もちろん図 1(A)，(B) 以外の箇条書きから「会社」の代わりに「車」を上位語として獲得することもありえる．しかし，実際には「トヨタ」を含む箇条書きは「ホンダ」や「ニッサン」を含む可能性が高く，多かれ少なかれ箇条書きは類似すると考えられる．そのため，このことと AHRAI が箇条書きの要素に関連した文書中に現れる単語の分布をもとに上位語を求めることを考え合わせると，「車」が「トヨタ」の上位語であることを AHRAI により認識できる可能性は低い．

その一方で，HEAIH は箇条書きの表題を利用することで，「トヨタ」，「ホンダ」，「ニッサン」の各表現が，「会社」と「車」の 2 つの語を上位語として持つことを認識する．HEAIH は，下位語を獲得したいと考えている上位語が与えられた時，その上位語を表題（より正確には表題の候補）として含むような HTML 文書を発見し，その中から表題と対応のとれている箇条書きを抽出する．そして，上位語を含む表題と，その表題と対応している箇条書きから上位下位関係を獲得する．しかし，ここで問題となるのは，どのようにして表題と箇条書きが対応しているかどうかを正確に認識するかということである．例えば，箇条書きと表題の対応関係を調べるために，HTML 文書のソースコード中に現れる両者間の距離（文字数）を用いる方法が考えられるが，後の実験で示すように，このような単純な方法ではあまりよい結果が得られない．この理由としては，デザインの凝った Web ページが非常に多いことが挙げられる．これらの Web ページはブラウザを通して見た時の表題と箇条書きの距離と，ソースコード中での距離が必ずしも一致していなかったり，表題と箇条書きの間に，Web ページのメニューなどの余計な文字列が挿入されているためである．HEAIH では，この問題を解決するために，AHRAI が箇条書きに対して上位語を獲得する際に出力する上位語候補のリストを利用する．そして，表題と箇条書きが同一の HTML 文書中に現れた時，その箇条書きから生成された上位語候補リストの上位を表題の部分文字列が占めているならば，その表題と箇条書きは対応していると仮定し，表題と箇条書き間の対応の度合いを計算する．この AHRAI の出力する上位語リストのトップだけでなく，上位の要素を考慮に入れるという点が，AHRAI では獲得できない上位語を HEAIH で獲得することを可能にする．つまり図 1 に示す箇条書きに対し，AHRAI は (A)，(B) ともにトップの上位語候補として，例えば「会社」を生成するが，仮に「車」がリストの上位に含まれていれば，HEAIH では (B) の箇条書きに対しその上位語として「車」を獲得することが可能になる．

以下本稿では，2 節で既存手法である AHRAI の概要について述べ，3 節にて本研究で提案する HEAIH について説明する．そして，実際の HTML 文書を利用して行った他の手法との比較実験について 4 節で触れ，5 節で本稿のまとめを行う．

2　先行研究: AHRAI

AHRAI は，以下の 3 つの仮説に基づき HTML 文書中に現れる箇条書きから上位下位関係を獲得する．

仮説 A　HTML 文書中に現れる同一の箇条書きに含まれる各表現は，共通した上位語を持ちやすい

仮説 B　共通の上位語を持つ下位語の集合が与えられた時，その共通する上位語は各下位語を少なくとも一つ含むような文書中に現れやすい

仮説 C　上位語と下位語は意味的に類似している

本稿では，同じ箇条書きに含まれる各表現を**下位語候補**，また下位語候補の集合を**下位語候補集合**と呼ぶ．図 1 の例でいえば，箇条書きの要素である

ステップ 1 HTML 文書からの下位語候補集合の獲得
ステップ 2 語の文書頻度を示す DF 及び IDF を用いた，各下位語候補に共通する上位語の選択
ステップ 3 上位語と下位語の意味的類似度に基づく，上位語候補と下位語候補集合のソート
ステップ 4 ヒューリスティックなルールによる，上位語候補と下位語候補集合の洗練

図 2　AHRAI による上位下位関係の獲得手順

「トヨタ」,「ホンダ」,「ニッサン」の各表現は下位語候補であり，これらは同じ下位語候補集合の要素とみなすことができる．

AHRAI は図 2 に示す 4 つのステップからなる．このステップ 1 から 3 は，先程示した 3 つの仮説にそれぞれ対応している．

ステップ 1 では，HTML 文書中に現れる各表現を囲んでいるタグに注目することで，下位語候補集合の獲得を行う．その詳細については，Shinzato ら [6] を参照されたい．残りのステップの概要に関しては，以下の通りである．

2.1　ステップ 2

ステップ 2 では，文書頻度などの統計的尺度を用いて，下位語候補集合の各要素に共通する上位語の獲得を行う．そのために，2 つの文書集合を準備する．1 つ目の文書集合は，大量の HTML 文書を Web よりランダムに収集し作成したもので，これを**大域的文書集合**と呼び，単語の一般的な文書頻度を求める際に使用する．続いて 2 番目の文書集合は，下位語候補集合中の各要素を検索語とし，既存の検索エンジン[2]より，1 検索語あたり 100 件の HTML 文書を収集し生成するもので，これを**局所的文書集合**と呼ぶ．この文書集合は，その中に含まれている名詞と下位語候補との関連の強さを測る際に用いる．

以下ステップ 1 で獲得された下位語候補集合を C，C について生成した局所的文書集合を $LD(C)$，大域的文書集合を G で表す．また，上位語候補になりえる名詞の集合を N [3]とする．この時，C に対する上位語候補 $h(C)$ は以下の式で計算される．

$$h(C) = \mathbf{argmax}_{n \in N}\{hS(n, C)\}$$

$$hS(n, C) = DF(n, LD(C)) \cdot IDF(n, G)$$

ここで，$DF(n, D)$ は，文書集合 D 中における名詞 n の文書頻度を求める関数であり，$IDF(n, G)$ は，$\log(|G|/DF(n, G))$ で定義される名詞 n の IDF 値を求める関数である．また，関数 hS は先述した仮説 B に基づき，局所的文書集合中に頻繁に現れ，かつ，大域的文書集合中にはあまり現れない名詞ほど大きな値を返す．

2.2　ステップ 3

ステップ 3 では，仮説 C に従い，上位語候補と下位語候補集合間の意味的類似度を考慮することで，上位語ではないが各下位語候補と関連の強い語を上位語として獲得している組の削除を行う．そのため，ステップ 2 までで獲得された上位語候補と下位語候補集合の組を，上位語候補と下位語候補集合の意味的類似度に従ってソートする．これにより，後述するステップ 4 を適用後，その上位幾つかを最終的な出力とすることで，誤った上位語が獲得されている組を出力結果から排除することが可能になる．上位語ではないが関連の強い語としては,「トヨタ」,「ホンダ」,「ニッサン」からなる下位語候補集合に対しては，例えば「価格」が考えられる．AHRAI では，このような語は，各下位語候補との意味的類似度が相対的に低いと考える．この例でいえば,「価格」と「ホンダ」,「トヨタ」,「ニッサン」の意味的類似度は低いと考えられるため，低く順位付けされることが期待でき，最終的な出力結果からは削除される．その一方で，適切な上位語は各下位語候補と意味的に類似していると考えられる．例えば,「ホンダ」,「トヨタ」,「ニッサン」に対し,「会社」という語は意味的に類似しているといえる．そのため，このような組は高く順位付けされやすく，最終的な出力結果に含まれる．

AHRAI では，上位語候補 n と下位語候補集合 C の間の意味的類似度 $sim(n, C)$ を，コサイン尺度を用い以下のように計算する．

$$sim(n, C) = (ho(C) \cdot hy(n))/(|ho(C)||hy(n)|)$$

ここで，$ho(C)$ は下位語候補集合中の要素全体の係り受け関係を表したベクトル（以下，このベクトルを**係り受けベクトル**と呼ぶ）であり，$hy(n)$ は上位語候補 n の係り受けベクトルである．全ての助詞を $\{p_1, \cdots, p_l\}$，動詞の集合を $\{v_1, \cdots, v_o\}$ で表した時，下位語候補集合 C 全体及び，上位語候補 n の持つ係り受けベクトル $ho(C)$，$hy(n)$ は，以下のように定義される．

$$ho(C) = \langle f_h(C, p_1, v_1), \cdots, f_h(C, p_l, v_o)\rangle$$

$$hy(n) = \langle f(n, p_1, v_1), \cdots, f(n, p_l, v_o)\rangle$$

ここで，$f_h(C, p, v)$ は，局所的文書集合中において，下位語候補集合 C の要素が助詞 p を介して動詞 v に係る頻度を表している．同様に，$f(n, p, v)$ は，大量の文書集合から学習した，上位語候補 n が助詞 p を介して動詞 v に係る頻度を表している．

[2]検索エンジンには goo(http://www.goo.ne.jp/) を用いており，検索時に，検索語の両端を “” ではさみ，検索語が自動的に分割されないようにしている．

[3]新聞記事 33 年分中に頻繁に現れる名詞 37,639 個からなる集合を N としている．

ルール 1 獲得された上位語候補を検索語として検索エンジンに問い合わせた結果得られるヒット件数が，各下位語候補を検索語として得られたヒット件数の総和よりも少ない場合，その上位語候補と下位語候補集合の組は削除する．

ルール 2 獲得された上位語候補が，下位語候補集合のいずれかの要素の部分文字列として現れていた場合，以下の条件に当てはまる上位語候補と下位語候補集合の組は削除する．

- 上位語候補が下位語候補の末尾以外の場所で部分文字列として現れている．
- 下位語候補集合の半分以上の要素で上位語候補が末尾に現れていない．

ルール 3 上位語候補が地名を表す語である場合，上位語候補を「地名」に変更する．

図 3　AHRAI で用いているヒューリスティックなルール

ステップ 2 までで得られた上位語候補と下位語候補集合の組を $\{\langle h(C_i), C_i\rangle\}_{i=1}^{m}$ で表す．ここで，C_i は下位語候補集合を，$h(C_i)$ は下位語候補集合 C_i に対してステップ 2 で計算された上位語候補を表している．ステップ 3 では，$\{\langle h(C_i), C_i\rangle\}_{i=1}^{m}$ を以下の値に基づきソートする．

$$sim(h(C_i), C_i) \cdot hS(h(C_i), C_i)$$

これにより，上位語候補と下位語候補集合が意味的に類似した組ほど高い順位を得ることが可能になる．後述するステップ 4 でヒューリスティックなルールを適用後，上位 k 組を最終的な出力結果とすることで，AHRAI は相対的に高い精度で上位下位関係の獲得を可能にしている．別の言い方をすれば，AHRAI は，残りの $m-k$ 組を誤った上位語候補が獲得されやすいという理由から出力せずに破棄する．

2.3　ステップ 4

ステップ 3 までで，上位語候補と下位語候補集合間の意味的類似度に基づいてソートされた，上位語候補と下位語候補集合の組が得られる．そのため，その上位幾つかを出力することで，ある程度の精度で単語間の上位下位関係を獲得することは可能である．しかし，図 3 に示すルールを適用し，ある組を削除したり，獲得された上位語候補を修正したりすることで，より高い精度で上位下位関係を獲得することができる．これらのルールの詳細については，Shinzato ら [6] を参照されたい．また，先述したように各ルールを適用後，上位 k 組が AHRAI の出力となる点に注意されたい．

3　提案手法：HEAIH

本節では，提案手法である HEAIH について述べる．HEAIH は，AHRAI の幾つかのステップを利用することで単語間の上位下位関係の獲得を行う．HEAIH は，l 個の上位語の集合 $X = \{x_i\}_{i=1}^{l}$ を入力として受取り（ここで x_i は上位語を意味している），各上位語に対する下位語を求める．そのために，まず上位語 x_i の下位語を要素とするような箇条書きを含む HTML 文書を Web よりダウンロードする．この操作は，予め用意しておいた言語パターンを利用し，与えられた上位語を含んだ箇条書きの表題として現れやすい表現を生成し，この表現を含む HTML 文書を既存の検索エンジンを利用して収集することで行う．続いて，AHRAI のステップ 1, 2 を適用することで，先程ダウンロードした文書集合中から抽出される下位語候補集合それぞれについて，上位語候補リストを生成する．この上位語候補リスト中の各語は，ステップ 2 で計算された hS の値に基づき，降順にソートされている．

“X(の) 一覧”，“X(の) ガイド”，“X(の) カテゴリ”，“X(の) ランキング”
“X(の) リスト”，“X(の) 投票”，“X(の) メニュー”，

1) X の部分には上位語が入る．2) (の) は省略可．

図 4　表題を生成するパターン

HEAIH では，与えられた上位語 x_i の部分文字列の多くが，生成された上位語候補リストの上位に含まれていれば，その上位語 x_i は，上位語候補リストを生成した下位語候補集合に対して妥当な上位語であると仮定する．そのため，与えられた上位語 x_i と下位語候補集合の各組に対するスコアを計算するが，これは上位語 x_i の部分文字列が，上位語候補リストの上位に含まれていればいるほど，大きな値をとるようにする．そして，このスコアに基づいて各組をソートし，上位 k 組を最終的な出力結果として獲得する．

より具体的には，HEAIH は以下に示す A から E の 5 つのステップからなる．

ステップ A　上位語 x_i に対し，図 4 に示すパターンを用いて，“x_i の一覧” のような表題として良く使われる表現の集合 $Hd(x_i)$ を生成する．ついで，既存の検索エンジンを用い，生成された $Hd(x_i)$ の各要素のヒット件数を調べ，最もヒット件数の多い表現を求める．そして，その表現について，検索エンジンより得られる一定数の HTML 文書を収集する．

ステップ B　AHRAI のステップ 1 を利用し，ステップ A で収集した HTML 文書集合から下位語候補集合を抽出する．ここで，ステップ B を適用した結果を $B(X) = \{\langle x'_h, C_h\rangle\}_{h=1}^{m}$ と表す．ここで，x'_h は与えられた上位語を，C_h は x'_h について収集した HTML 文書集合中から獲得された下位語候補集合をそれぞれ表す．

ステップ C AHRAIのステップ2を，ステップBで抽出した下位語候補集合 C_h に適用し，hS の値に基づいてソートされた上位語候補，上位 p 語からなる上位語候補リストを獲得する．以降，このリストを $HCList(C_h)$ で表す．

ステップ D 与えられた上位語と下位語候補集合の組 $B(X) = \{\langle x'_h, C_h\rangle\}_{h=1}^{m}$ を以下の式で与えられる hSC の値に基づいてソートする．

$$hSC(x'_h, C_h) = sim(x'_h, C_h) \cdot \sum_{j=1}^{p}\{sub(x'_h, jth(HCList(C_h), j)) \cdot hS(jth(HCList(C_h), j), C_h)\}$$

ただし，

$$sub(x, y) = \begin{cases} 1 & y \text{ が } x \text{ の部分文字列である時} \\ 0 & \text{上記以外の時} \end{cases}$$

であり，また，$jth(list, j)$ は上位語候補リスト $list$ 中の j 番目の要素を表す．hSC の値は，おおまかに言えば与えられた上位語 x'_h の部分文字列の中で，上位語候補リスト $HCList(C)$ の上位 p 語に含まれているものの持つ hS の値の総和であるが，与えられた上位語と下位語候補集合の意味的類似度の値 $sim(x, C)$ [4]も考慮している点に注意されたい．また，本研究では，経験的に $p = 10$ としている．

ステップ E ステップDで獲得された，hSC のスコアに基づきソートされた上位語と下位語候補集合の組に対し，図3に示したルール1と2を適用する．その後，残った上位語と下位語候補集合の組のうち上位 k 組を上位下位関係として出力する．後述する実験では，$|B(X)|$ が2,000程度の時，$k = 200$ としている．

4 実験

4.1 設定

HEAIHを評価するため，適当な上位語の集合を準備する必要がある．しかし，実際には上位語となり得ない名詞が数多く存在するため，上位語になり得る名詞だけを獲得することはそれほど簡単ではない．本研究では，“X 一覧”，“X の紹介”，“歴代の X”，“有名 X” という4つパターンを用意し，これらパターンの X の部分に当てはまる名詞列は上位語として適切であるという仮定を設け，コーパス中から X の部分に当てはまる名詞列を抽出し，それらをHEAIHに与える上位語の集合とした．実際に，Webよりダウンロードした 1.00×10^6 件のHTML文書（1.26 GB，タグなし）に対し，上記のパターンを適用した結果，8,752個の表現が得られ，その中から無作為に200個選びだして実験に用いた[5]．しかし，このようなパターンを用いて名詞の選別を行っているにも関わらず，結果的に上位語として不適切な語が抽出されてしまった．この不適切な語の中には，語が使われている状況をかなり考慮しなければ，上位語として考えにくいものもある．その例として「おすすめ商品」が挙げられる．「おすすめ商品」は，誰がすすめているのかがわからなければ，その下位語を決定することができないため，上位語として不適切であると考えられる．本研究では，不適切な語について獲得された上位下位関係は，誤ったものとして評価した．

ついで，各上位語ごとに最大で25件のHTML文書を検索エンジン goo [6] より収集し，それらに対しステップBを適用することで，3,211個の下位語候補集合を獲得した（この時，下位語候補集合の要素数が30を越えるものに関しては，後続の処理に多大な時間を要するという理由により，実験データから省いている）．そして，3,211個の下位語候補集合から，以下の手順により2,034個選びだし，これらに関して評価実験を行った．まず，各上位語に関して獲得された下位語候補集合を，HTML文書のソースコード中における下位語候補集合と上位語の間の距離（文字数）に従って昇順にソートし，その上位65 %の下位語候補集合を各上位語に関して獲得した．これにより，上位語とあまりにも遠く離れた場所から獲得された下位語候補集合は削除され，およそ2,000個程度の下位語候補集合について評価実験を行うことができる．ここで，2,000という数字は，以前我々がAHRAIの評価を行う際に用いた下位語候補集合の数である．

先述したように，HEAIH及びAHRAIは，局所的文書集合と大域的文書集合の2つの異なる文書集合を必要とする．本研究では大域的文書集合として，上位語を獲得する際に用いた 1.00×10^6 件のHTML文書を利用した．一方，局所的文書集合としては，下位語候補集合中の各要素に関して，先程と同様，検索エンジン goo で検索し，得られた検索結果の上位100件のHTML文書をダウンロードして作成したものを用いた．また，上位語と下位語候補集合の意味的類似度を計算するために必要な係り受けベクトルを得るため，本研究では，5.72×10^6 件のHTML文書を用いた．係り受けベクトルを求めるため，HTML文書を既存の構文解析器 [9] [7]を

[4]本研究では，Webより収集した大量の文書集合中に含まれる名詞のうち，ある頻度以上で現れているものに対して，係り受けベクトルを獲得している．そのため，文書集合に含まれていなかったり，低頻度でしか現れていない名詞あるいは複合語については，係り受けベクトルを獲得できておらず，意味的類似度 sim を計算することができない．しかし，そのような場合は，名詞もしくは複合語の接尾辞の中で，係り受けベクトルが獲得されている最長のもので代用している．

[5]実際は，上記のパターンに4回以上適合した869個の表現の中から，無作為に100個の表現を抜き出し，さらに残りの7,883個の表現の中からも，無作為に100個の表現を抜き出し，それらを合わせて用いている．

[6]http://www.goo.ne.jp/

[7]金山ら [9] では素性構造の単一化を行っているが，本実験で

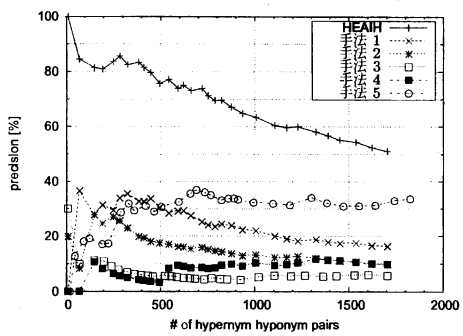

図 5　上位下位関係獲得の精度

利用して解析し，HTML 文書中に含まれる名詞と動詞の共起関係を調べた．

4.2 HEAIH による上位下位関係の獲得の性能評価

最初の評価実験として，HEAIH によりどのくらいの精度で上位下位関係が獲得できるかどうかを評価した．3 節で述べたように，HEAIH は上位語と下位語候補集合の組，上位 200 組を，上位下位関係として出力する．上位下位関係獲得精度と最終的に出力する上位下位関係数の関係を図 5 に示す．図中で "HEAIH" と示されたグラフが，HEAIH の上位下位関係の獲得精度である．この時，上位下位関係獲得の精度は次のようにして求めている．まず，ソートされた上位語と下位語候補集合の組のうち，上位 j 組を取り出す．そして，その中で適切な上位下位関係が獲得されている割合を計算し，それを精度とした．図 5 中のグラフの横軸は，ソートされた上位語と下位語候補集合の組の上位 j 組中に含まれる下位語候補の数を示しており，縦軸は適切な上位語が獲得された下位語候補の割合を示している．つまり，グラフの各線は以下の式に従って描画されている．

$$\langle \sum_{h=1}^{j} |C_h|, \frac{\sum_{h=1}^{j} correct(C_h, x'_h)}{\sum_{h=1}^{j} |C_h|} \rangle$$

ここで j は $1 \leq j \leq 200$ であり，$|C_h|$ は下位語候補集合 C_h の要素数である．さらに $correct(C_h, x'_h)$ は上位語 x'_h が適切な上位語である下位語候補集合 C_k 中の要素数を表している．

評価実験に用いた 2,034 組の上位語と下位語候補集合の組には，16,433 個の上位下位関係が含まれているが，そのおよそ 5 %にあたる，上位 799 個の上位下位関係（上位語の異なり数は 21 個）を最終的な出力とすると，その精度は約 70 %，およそ 10 %にあたる，上位 1,602 個（異なり数は 45 個）について

用いたバージョンでは，単一化の近似だけを行っている．

表 1　実際に獲得された上位下位関係の例

与えた上位語	獲得された下位語候補集合
皇帝	世宗*, 始祖*, 敬宗*, 統宗*, 高祖*, 恭宗*
福祉施設	身体障害者授産施設*, 身体障害者療護施設*, 重度身体障害者更生施設*
健康食品	ルイボスティー*, プーアル茶*, シモン茶*, グルコケア*, 紫イペ-*
占いサイト	占いカフェ*, 占い比較市場* 矢萩予言研究所*, うらないサーチ*
小説家	武揚伝, 田端文士村, 山布院心中事件

研究室 (34)*, 健康食品 (18)*, 福祉施設 (13)*, 機能 (12), 都市公園 (10)*, 店 (10)*, 皇帝 (7)*, 地区 (6)*, 事業 (6), 遺産 (6)*, 取り扱い商品 (5), 参加企業 (5), 作品 (5)*, パーツ (5), 日本三大○○ (4), 小説 (4)*, 部活動 (3)*, 占いサイト (3)*, 事業制度 (3), タイムアタック (3), コマンド (3), 注目商品 (2), 生産者 (2), 詩 (2)*, 市 (2)*, 高山植物 (2)*, チーム名 (2)*, サイドビジネス (2), お仕事 (2), 物件 (1), 日本語版 (1), 動物 (1)*, 専門 (1), 紹介 (1), 小説家 (1)*, 質問 (1), 資料 (1), 在宅ビジネス (1), 学童クラブ (1)*, 会場 (1) 駅名 (1)*, マルチメディア科 (1), パワーストーン (1)*, バンド (1)*, シェフ (1)*, ゲームソフト (1)*, キャラ (1)*, アイドル (1)*

図 6　HEAIH の出力結果中に含まれる上位語の一覧

は，約 50 %の精度で上位下位関係を獲得できている．この精度は十分に高いとはといえないが，誤りの多くは，先述したような不適切な上位語が入力として与えられたことに起因していると考えられる．実際に後述する実験では，不適切な上位語を取り除いた場合についても評価を行っており，その場合の精度は向上している．

次に，HEAIH により実際に獲得された上位下位関係を表 1 に示す．表 1 中で，"*" のついた下位語候補は，与えられた上位語に対して適切な下位語であることを意味しており，その評価基準は Miller ら [4] の定義に従い，"下位語候補は一種の上位語である" が成り立つがどうかで，適切であるかどうかを判断をしている．

獲得された 200 個の下位語候補集合を見てみると，119 件の異なる HTML 文書からそれぞれ獲得されており，1 件の HTML 文書からは最大で 7 個の下位語候補集合が獲得されていた．また，200 組の上位語と下位語候補集合中に含まれている上位語の異なり数は 48 であった．これら 48 種類の上位語と，その上位語について獲得された下位語候補集合の数（括弧内）を図 6 に示す．

上位語「下位語」, 下位語, .* 以外の .* 上位語,
下位語 .* のような .* 上位語,
下位語 .* に似た .* 上位語,
下位語 .* など (、| の)? 上位語,
下位語 .* と呼ばれる .* 上位語,
下位語 .* と (い | 言) う .* 上位語,
下位語 .* (ら | たち) .* 上位語

上位語と下位語は「」や“”によって囲まれていても構わない.

図 7　比較実験で用いている構文パターン

4.3　既存手法との比較実験

続いて，以下の 5 種類の手法と HEAIH との比較実験を行った．各手法については次のとおりである．

手法 1 与えられた上位語が下位語候補の接尾辞となっていれば，それらを上位下位関係として獲得する方法を手法 1 とする．これは，下位語の接尾辞は妥当な上位語になりやすいという日本語の特性を利用している．例えば，「プロ野球選手」や「冷凍食品」の上位語としては，その接尾辞である「選手」や「食品」が妥当であると考えられる．

手法 2 ステップ C で生成した局所的文書集合中から，先行研究で用いられている構文パターンにより上位下位関係を抽出する方法を手法 2 とする．このとき，構文パターンは今角 [8]，安藤ら [7] で使われたものをもとにしている（図 7）．本研究で用いているパターンは正規表現であるため，余計な上位下位関係を生成してしまうかもしれないが，本研究で用いたパターンは，従来手法で用いられている構文パターンによって獲得される上位下位関係を獲得し損ねることはない．

手法 3 既存の検索エンジンと構文パターンを用いて上位下位関係を獲得する方法を手法 3 とする．基本的には，手法 3 でも手法 2 と同じ構文パターンを用いる．しかし，検索エンジンでは正規表現が利用できないため，図 7 に示した各構文パターンから “.*” を取り除き，“|” の部分は単純な文字列に展開したものを用いる．さらに，“上位語「下位語」” パターンについては，検索エンジンで “「」” を正しく扱えないため，このパターンを用いた上位下位関係の獲得は行っていない．

手法 4 我々が以前に提案した手法 AHRAI により，上位下位関係を獲得する方法を手法 4 とする．

手法 5 与えられた上位語を含む表題と，その上位語について獲得された下位語候補集合との，HTML 文書中における距離（文字数）にのみ注目して上位下位関係を獲得する方法を手法 5 とする．この手法では，「表題は下位語候補集合の前に現れる」という仮定に基づき，最もヒット件数の多かった $Hd(x)$ [8] の要素に関して収集した文書中において，その $Hd(x)$ の要素と下位語候補集合間の距離を計算する．そして，計算された距離に基づいて，与えられた上位語と下位語候補集合の組を昇順にソートし，その上位 200 組を上位下位関係として出力する．

実験では，上述した 5 つの比較手法が，HEAIH により獲得された妥当な上位下位関係を獲得することができるかどうかという観点で評価を行った．つまり，同じ文書集合を用いた時，HEAIH によって獲得された妥当な上位下位関係が，比較手法によりどのくらい獲得されるかを計算したことになる．ここで，手法 5 以外の 4 手法は，HEAIH により獲得された上位語と下位語候補集合 200 組に対して，それぞれの手法を適用している点に注意されたい．

比較実験の結果を図 5 に示す．手法 1 から 4 の各線は，HEAIH により獲得された上位語と下位語候補集合の組に対して，各手法を適用した時の精度を示している．一方で，手法 5 で獲得された上位下位関係については，ステップ B 後に獲得される上位語と下位語候補集合の組，2,034 組から，上述した方法で得られた上位下位関係の精度を示している．図 5 より，比較手法では獲得できないような上位下位関係を HEAIH では多数獲得できていることがわかる．つまり，比較的少量の文書を使用した時（下位語 1 つあたり 100 件の HTML 文書），我々が以前に提案した AHRAI も含め，既存手法では獲得できないような上位下位関係を HEAIH では獲得できているといえる．また，手法 3 と比べることで，検索エンジンを利用し，従来手法で用いられている構文パターンを検索する方法では見つけることのできない上位下位関係も獲得できていることがわかる．これは，仮に非常に大量の文書を使用しても構文パターンでは見つけることのできない上位下位関係を，HEAIH では獲得できるという可能性を強く示唆しているといえる．

4.4　上位語の偏り及び，不適切な上位語を取り除いた場合の比較実験

最後に，HEAIH と手法 1 から 5 を前節の実験とは若干異なる設定で，比較実験を行い再評価した．図 6 に，HEAIH の出力結果に含まれる上位語と，各上位語が持つ下位語候補集合の数が示されているが，注意して見ると「研究室」に関しては 34 個の下位語候補集合を獲得しているにも関わらず，「動物」に関しては 1 個と与える上位語によって獲得される下位語候補集合の数に偏りが見られる．そこで，よ

[8] ここで $Hd(x)$ は，3 節同様，与えられた上位語 x を含む，箇条書きの表題となりやすい表現の集合を表す．

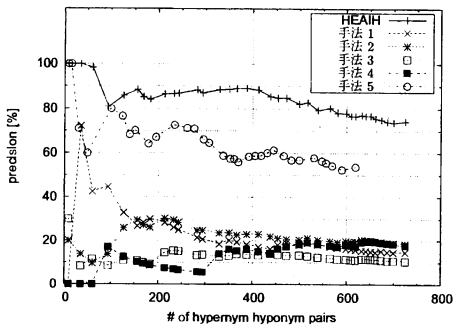

図 8　偏りの少ないデータを用いた評価実験結果

り偏りの少ないデータを用いて再度比較実験を行った．そのために，HEAIH の出力結果から，それぞれの上位語について上位語と下位語候補集合の組，上位 5 組を取りだし，それ以外の組に関しては出力結果から削除した．つまり，今回評価した上位語と下位語候補集合の組の中には，1 つの上位語につき最大で 5 個の下位語候補集合しか含まれていない．これにより，獲得された 200 組の上位語と下位語候補集合の組の中に，頻繁に現れる上位語による影響を軽減することができる．

さらに，評価データの中から問題となりそうな上位語の削除も行った．図 6 において "*" のついていない上位語は，今回評価データに含めなかったものである．本研究では，その下位語として固有名詞，動物などの品種名，商品名を持つ上位語を評価データに残した[9]．また，先述した上位語の使われている状況がわからない状態では下位語を決めることのできない上位語についても評価データから削除した．さらに，一般的すぎるという理由から上位語「物件」も削除した．その結果，評価データとして 73 組の上位語と下位語候補集合の組が残った．

このデータを用いた比較実験の結果を図 8 に示す．図より，このような環境においても，HEAIH は比較手法では獲得できないような上位下位関係を多く獲得できていることがわかる．HEAIH の獲得精度は，機械的に上位語を選択し，入力として与えた場合に比べ若干高く，これは出力の選択方法及び，入力として与える上位語の選択方法をより吟味することによって，性能の向上が図れる可能性を示唆しているといえる．

[9] この条件は明らかに本来の上位下位関係よりも制限が厳しく，直感的には上位語として見なしても構わない語についても評価データから削除してしまっている．そのような語の例として，「お仕事」や「事業」が挙げられる．本研究では，できるだけ単純な条件で，上位語かどうかを判断することが難しい場合を減らすように努めた結果，この条件を設けた．4.2，4.3 節で述べている実験では，この条件により削除された「お仕事」や「事業」のような上位語も，適切な上位語として扱われていることに注意されたい．

5　まとめ

本稿では，HTML 文書中に現れる箇条書きとその表題を用いて，予め指定された語に対する下位語を獲得する方法について述べた．この手法は，我々が以前に提案した，HTML 文書中に含まれる箇条書きを利用して上位下位関係を獲得する手法を拡張したものである．比較実験により，比較的少量の文書を使用した時（下位語 1 つあたり 100 件の HTML 文書），既存の手法では獲得できないような上位下位関係を，提案手法では獲得できることが確認された．また，検索エンジンを利用し，従来手法で用いられている構文パターンを検索する方法では見つけることのできない上位下位関係も獲得できた．これはつまり，仮に非常に大量の文書を用いても，構文パターンでは抽出できない上位下位関係を提案手法では獲得できる可能性を強く示唆しているといえる．

また，我々の最後の実験では，出力の選択方法及び，入力として与える上位語の選択方法をより吟味することによって，性能の向上が図れる可能性が示唆された．特に，前述した不適切な上位語を予め排除する手法は今後の課題の 1 つである．

参考文献

[1] Caraballo, S. A.: Automatic construction of a hypernym-labeled noun hierarchy from text, *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pp. 120–126 (1999).

[2] Fleischman, M., Hovy, E. and Echihabi, A.: Offline Strategies for Online Question Answering: Answering Questions Before They Are Asked, *Proceedings of the 41th Annual Meeting of the Association for Computational Linguistics*, pp. 1–7 (2003).

[3] Hearst, M. A.: Automatic acquisition of hyponyms from large text corpora, *14th International Conference on Computational Linguistics*, pp. 539–545 (1992).

[4] Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D. and Miller, K. J.: Introduction to WordNet: An on-line lexical database, *Journal of Lexicography*, pp. 235–244 (1990).

[5] Morin, E. and Jacquemin, C.: Automatic acquisition and expansion of hypernym links, *Computer and the Humanities 2003* (2003). forthcoming.

[6] Shinzato, K. and Torisawa, K.: Acquiring hyponymy relations from web documents, *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics 2004*, pp. 73–80 (2004).

[7] 安藤まや, 関根聡, 石崎俊: 定型表現を利用した新聞記事からの下位概念単語の自動抽出, 情報処理学会研究報告 2003-NL-157, pp. 77–82 (2003).

[8] 今角恭祐: 並列名詞句と同格表現に着目した上位下位関係の自動獲得, 修士論文, 九州工業大学 (2001).

[9] 金山博, 鳥澤健太郎, 光石豊, 辻井潤一: 3 つ組・4 つ組モデルによる日本語係り受け解析, 自然言語処理, Vol. 7, No. 5 (2000).