HTML 文書からの単語間の上位下位関係の自動獲得

新里　圭司[†]　鳥澤 健太郎[†]

本稿では，HTML 文書より単語間の上位下位関係を自動獲得する手法を提案する．従来より，単語間の上位下位関係は自然言語処理において重要な知識であると見なされており，多くの自動獲得手法が提案されてきた．しかし，それらの多くは名詞句の並置などの文の表層的なパターンに注目するものであり，広範な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る．そのため本稿で提案する手法は，これらとは異なるアプローチをとる．より具体的には，1) HTML タグにより与えられる HTML 文書の構造，2) 従来より情報検索などで使われてきた *df*，*idf* などの統計量，3) 大量のテキストから獲得した名詞と動詞の係り受け関係，4) 予備実験より得られた知見に基づくヒューリスティックなルール，の４つの要素を組み合わせることで，様々な単語間の上位下位関係を自動的に獲得することを目指す．

キーワード:　知識獲得，上位語，下位語，統計的自然言語処理，*World Wide Web*

Automatic acquisition of hyponymy relations from HTML documents

This paper describes an automatic acquisition method for hyponymy relations. Hyponymy relations play a crucial role in various natural language processing systems, and there have been many attempts to automatically acquire the relations from large-scale corpora. Most of the existing acquisition methods rely on particular linguistic patterns, such as juxtapositions, which specify hyponymy relations. Our method, however, does not use such linguistic patterns. We try to acquire hyponymy relations from four different types of clues. The first is repetitions of HTML tags found in usual HTML documents on the WWW. The second is statistical measures such as *df* and *idf*, which are popular in IR literatures. The third is verb-noun co-occurrences found in normal corpora. The fourth is heuristic rules obtained through our experiments on a development set.

KeyWords: *Knowledge acquisition, Hypernym, Hyponym, Statistical Natural Language Processing, World Wide Web*

1　はじめに

近年，膨大な量の文書が計算機で扱えるようになり，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．しかし，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*），同義関係（*synonymy relation*），包含関係（*part-whole relation*）な

† 北陸先端科学技術大学院大学 情報科学研究科, School of Information Science, Japan Advanced Institute of Science and Technology

どの知識がまだまだ不足しており，このような知識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，WWW 上に大量にある HTML 文書から広範な単語間の上位下位関係を自動的に獲得する手法について述べる．Miller ら (Miller, Beckwith, Fellbaum, Gross, and Miller 1990) 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* ”が言える時であると定義[1]されており，本研究でもこの定義に従う．また単語 B が単語 A の下位語であるということを次の形式で記述する．

$$\text{HYPONYM}(\text{A}, \text{B})$$

例えば，茄子と野菜，秋刀魚と魚，冷蔵庫と機械の間には次のような関係が成り立つ．

HYPONYM(“野菜”，“茄子”)
HYPONYM(“魚”，“秋刀魚”)
HYPONYM(“機械”，“冷蔵庫”)

このような単語間の上位下位関係は種々の自然言語処理アプリケーションにおいて有用である．例えば，情報検索における検索質問拡張では，検索語に加え，検索語の上位語，下位語，類義語を付け加えて検索することで，再現率が向上することが報告されている (Mandala, Tokunaga, and Tanaka 1998)．これは，特許検索等の検索に漏れがあっては困るようなシステムに，単語の上位下位関係が有効であること示している．また，QA の分野においても,「ニューヨーク市の市長は誰か」や,「ナディア・コマネチは誰か」といった類の質問に，単語間の上位下位関係を利用して答えるといった研究が行われている (Fleischman, Hovy, and Echihabi 2003)．

本研究で，WWW 上の HTML 文書を対象としたのは，新聞記事などの他のコーパスと比べ，1) 量が豊富にある，2) 新規に「発明された」語や表現が素早く掲載される，3) HTML 文書製作者の何らかの意図に基づいて文書がタグ付けされている，といった特徴を HTML 文書は持っており，広範な単語間の上位下位関係を獲得するためにその特徴が使えるのではないかと考えたためである．

後述するように，従来より研究されてきた単語間の上位下位関係の獲得手法は，コーパス中から構文パターン（*lexico–syntactic pattern*）のマッチングにより獲得するものであったため，大量かつ幅広い単語間の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．そのため，本研究では従来法で用いられてきたような構文パターンによる獲得は行わず，それ以外の上位下位関係の特性を捉える手がかりを用いることで獲得を試みる．具体的には，タグにより与えられる HTML 文書の構造，情報検索などの分野で用いられる *df* や *idf* などの統計量，新聞記事から収集した名詞と動詞の係り受け関係，予備実験より得た知見に基づき作成したヒューリス

1 より正確には “*A concept represented by a lexical item L_0 is said to be a hyponym of the consepct represented by a lexical item L_1 if native speakers of English accept sentences consutructed from the frame* **An L_0 is a (kind of) L_1**.” と定義されている．

ティックなルール，の 4 つの異なる要素を組み合わせることで上位下位関係の獲得を行う．実際に WWW より収集してきた約 87 万件の HTML 文書に対して本手法を適用することで，下位語の集合（の候補）を約 9 万個獲得することができた．そして，その中からランダムに抽出した下位語の集合 2,000 個について評価を行った．評価は，2,000 個の集合に含まれる約 14,000 個の下位語に対して, 上位語を獲得し，結果として得られた上位語，下位語の対を後述の方法にしたがって順序づけして行った．その結果，全体の約 3.6％にあたる上位 501 個の上位下位関係については 85％，全体の約 5％にあたる上位 700 個については 75％，約 10％にあたる 1,400 個については 60％程度の精度で正しい上位下位関係を獲得することができた．

以下，本稿では 2 節で先行研究について触れ，3 節で本研究で提案する手法について説明する．4 節と 5 節で提案手法の性能を評価し，6 節で本稿のまとめを行う．

2 先行研究

これまでにも単語間の上位下位関係の獲得について多くの研究が行われてきた. しかし，そのほとんどが，新聞記事などのコーパスから構文パターンのマッチングによって上位下位関係の獲得を行うものとなっている (Hearst 1992; Caraballo 1999; 今角 2001; Morin and Jacquemin 2003; Fleischman et al. 2003; 安藤, 関根, 石崎 2003)．例えば，Hearst は，単語間の上位下位関係を表す構文パターンとして，以下のような *such as* パターンを発見した．

$$such\ NP\ as\ \{NP,\ \}^*\ \{(or \mid and)\}\ NP$$

Hearst は *such as* パターンを，コーパス中の文に適用することで単語間の上位下位関係の自動獲得を行っている．例えば，

. . . works by such authors as Herrick, Goldsmith, and Shakespeare.

という文に対して *such as* パターンを適用すると

$$\text{HYPONYM}(\textit{“author”},\ \textit{“Herrick”})$$
$$\text{HYPONYM}(\textit{“author”},\ \textit{“Goldsmith”})$$
$$\text{HYPONYM}(\textit{“author”},\ \textit{“Shakespeare”})$$

という単語間の上位下位関係を得ることができる．

このような構文パターンを日本語の新聞記事に対して適用し，単語間の上位下位関係の自動獲得を試みたものとして今角 (今角 2001)，安藤ら (安藤他 2003) の研究がある．今角は構文解析の結果より得られる同格・並列表現を含む文に対し，表 1に示すような構文パターンを用いて上位下位関係の獲得を行っている．新聞記事 4 年分（232 万文）に対し実験を行ったところ約 15,000 件の上位語下位語対が獲得でき，そのうち 600 件について人手で評価を行い，精度は 77.2%であったと報告している．また安藤らは，大規模シソーラスを自動的に生成するための準

表 1　先行研究で用いられている構文パターン

今角が用いた構文パターン	安藤らが用いた構文パターン	
名詞句 A「**名詞句 B**」	**名詞句 A** など**名詞句 B**,	**名詞句 A** などの**の名詞句 B** *
名詞句 A など, **名詞句 B**	**名詞句 A** に似た**名詞句 B**,	**名詞句 A** のような**名詞句 B** *
名詞句 A などの**名詞句 B** *	**名詞句 A** 以外の**名詞句 B**,	**名詞句 A** という**名詞句 B**
名詞句 A のような**名詞句 B** *	**名詞句 A** と呼ばれる**名詞句 B**	

パターンの後ろに “*” がついているものは両方の研究で用いられているものである．

備として,「X などの野菜」といった構文パターンを用いて，連想概念辞書 (岡本 石崎 2001) に登録されている日常性の高い約 60 語についてその下位語の獲得を行っている．構文解析済みの新聞記事 6 年分に対して，表 1に示す 7 種類の構文パターンを適用することで，いずれのパターンについても約 60%から 85%程度（期待値[2]は 68.2%）の精度で正しい上位下位関係が獲得できたと報告している．一見，今角のものより精度が低いように思えるかもしれない．しかし，今角が用いているパターンに限ればその獲得精度の期待値は 81.3%であり，安藤らの方が若干高い．

これら，従来手法の問題点は，コーパス中に上位下位関係を表す構文パターンがそれほど頻繁に現れず，たとえ大量のテキストをもってきたとしても構文パターンに現れない単語や句が大量にある，といった点であり，大量かつ幅広い単語間の上位下位関係を獲得することが難しい．そのため，本研究では構文パターンによる獲得は行わず，3 節で述べるような構文パターン以外の上位下位関係の特性を捉える手がかりを用いることで獲得を試みる．さらに，今回行った比較実験により，我々の提案手法は，同一のコーパスを利用した場合，構文パターンで取得できないような上位下位関係を多数獲得できることが確認できた．その詳細については 5 節を参照されたい．

3　提案手法

3.1　概要

本研究では，以下に示す 3 つの仮説をたて単語間の上位下位関係の獲得に用いている．

仮説 1　HTML 文書中に現れる箇条書きやリストボックス，テーブルのセルなどの「繰り返し」の要素，あるいはその中に含まれる自然言語の表現は，意味的に類似しており共通の上位語を持ちやすい

仮説 2　共通の上位語をもつ下位語の集合が与えられた時，その共通の上位語は各下位語を（少なくとも一つ）含む文書に現れやすく，それ以外の文書には比較的現れにくい

仮説 3　上位語と下位語は意味的に類似しており，その類似性は上位語と下位語の持つ係り受け関係によって捉えることができる

2 論文 (安藤他 2003) では各パターンにより獲得できた下位語数とその精度しか報告されていない．そのため，ここで挙げた期待値は筆者が論文から求めた値である．

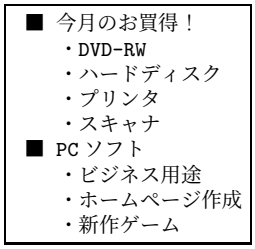

図 1　HTML 文書中に現れる箇条書きの例

そして，上の仮説に基づいた以下に示す 4 つのステップを経ることで単語間の上位下位関係の自動獲得を行う．ここに挙げたステップ 1，2，3 は上の仮説 1，2，3 とそれぞれ対応している．

ステップ 1　HTML 文書中のタグ情報に基づいた下位語候補集合の獲得

ステップ 2　*df*，*idf* などの統計量に基づく上位語候補の獲得

ステップ 3　上位語候補と下位語候補間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並べ替え

ステップ 4　ヒューリスティックなルールを用いた上位語候補と下位語候補集合の組の修正ならびに取捨選択

ここでステップ 4 は，上位下位関係獲得の精度を改善するために，ステップ 1，2，3 を通して獲得された上位下位関係をヒューリスティックなルールに従い修正，または削除するステップである．

本手法では，ステップ 1 において WWW より大量の HTML 文書をダウンロードし，その中から仮説 1 に従い同じリストの項目になっている表現や，同じテーブルの要素となっている表現を獲得する．例えば，図 1に示すような HTML 文書を考えた場合，ステップ 1 では次のようなパソコンの周辺機器とソフトウェアのジャンルからなる 2 つの集合を獲得する．

{DVD–RW，ハードディスク，プリンタ，スキャナ }
{ ビジネス用途，ホームページ作成，新作ゲーム }

本研究では，ステップ 1 で獲得された集合を**下位語候補集合**，集合の各要素を**下位語候補**と呼び，同じ集合内の下位語候補は共通の上位語（この例でいえば「機器」や「ジャンル」）を持つと考える．ここで “候補” と付いているのは，ステップ 1 で獲得される HTML 文書中の表現の集合が，必ずしも共通の上位語を持つとは限らないためである．

次いでステップ 2 では，従来より情報検索の分野などでよく用いられている *df* や *idf* といった統計量を利用し，ステップ 1 で獲得された各下位語候補に共通な上位語を獲得する．そのためにステップ 2 では，まず下位語候補を少なくとも 1 つ含むような文書を既存の検索エンジンを用いて WWW からダウンロードする．そして，ダウンロードした文書中に含まれる名詞のス

コアを計算し，スコアの最も高い名詞を上位語の候補として獲得する．本研究では，この獲得された名詞のことを**上位語候補**と呼ぶ．ここでも “候補” と付いているのは，ステップ 2 で獲得された名詞が最終的な上位語となるわけではなく，獲得された名詞の幾つかは後述するステップ 4 で修正される可能性があるためである．ステップ 2 で用いる名詞のスコアの計算式は，仮説 2 に基づき，ダウンロードした下位語候補を含む文書集合中の文書に現れやすく，他の文書に比較的現れにくい名詞ほど高いスコアを得るようにする．先程の例でいえば，DVD–RW やハードディスクを検索語として WWW よりダウンロードした文書群には，実際に多くの文書中に正しい上位語である「機器」が含まれ，さらにそれら以外の文書での「機器」の頻度は比較的低い．このことから,「機器」は高いスコアを得ることになる．

しかし，上位語ではないが DVD–RW やハードディスクと関連の強い名詞，例えば「データ」などの語も，DVD–RW などの表現を含む文書の多くに現れるため高いスコアを得てしまう．そこでステップ 3 では，このような上位語ではない名詞を誤って上位語候補として獲得している上位語候補と下位語候補集合の組を，最終的な出力結果から削除する．そのためステップ 3 では，仮説 3 に基づき，上位語候補と下位語候補の持つ係り受け関係から，両者間の意味的類似度を計算し，その値に従って上位語候補と下位語候補集合の組をソートする．上位語候補と下位語候補集合の組をソートすることにより，そのうち上位幾つかを最終的な出力結果とすることで，上位語候補と下位語候補に類似性の見られない組に関して削除することができる．例えば先程の例において，上位語候補として「データ」が獲得された場合,「データ」と DVD–RW, ハードディスク，プリンタ，スキャナは似た係り受け関係を持ちにくいため，類似性が弱いと考えられ，最終的な出力結果からは除かれる，

最後にステップ 4 として，予備実験より得た知見を基に作成したヒューリスティックなルールを，ステップ 1 から 3 までで獲得された上位語候補と下位語候補集合の組に対して適用し，上位語候補の修正や，上位語候補と下位語候補集合の組の削除を行う．本手法ではステップ 4 を適用後，残った上位語候補と下位語候補の組の中から，上位幾つかを最終的に獲得された上位下位関係として獲得する．

以上が，本研究で提案する構文パターンを用いずに単語間の上位下位関係を獲得する手法の概要である．以降本節では，各ステップについて説明する．

3.2　下位語候補集合の獲得 (ステップ 1)

ステップ 1 は，WWW より大量にダウンロードしてきた各 HTML 文書から，前述した仮説「**HTML 文書中に現れる箇条書きやリストボックス，テーブルのセルなどの「繰り返し」の要素[3]は，意味的に類似しており共通の上位語を持ちやすい**」に基づき，共通の上位語を持つであ

3 本研究では，後述のように「繰り返し」要素を特定の HTML タグによって同定しているわけではない．しかしながら，実際に獲得された繰り返し要素を見ると，他のページへのリンクを表す<A>タグ，文字の色や大きさを変更する<FONT>タグ，囲まれた文字がテーブルのセルの要素であることを意味する<TD>タグ，文字を太字に変更する<B>タグ，囲まれた

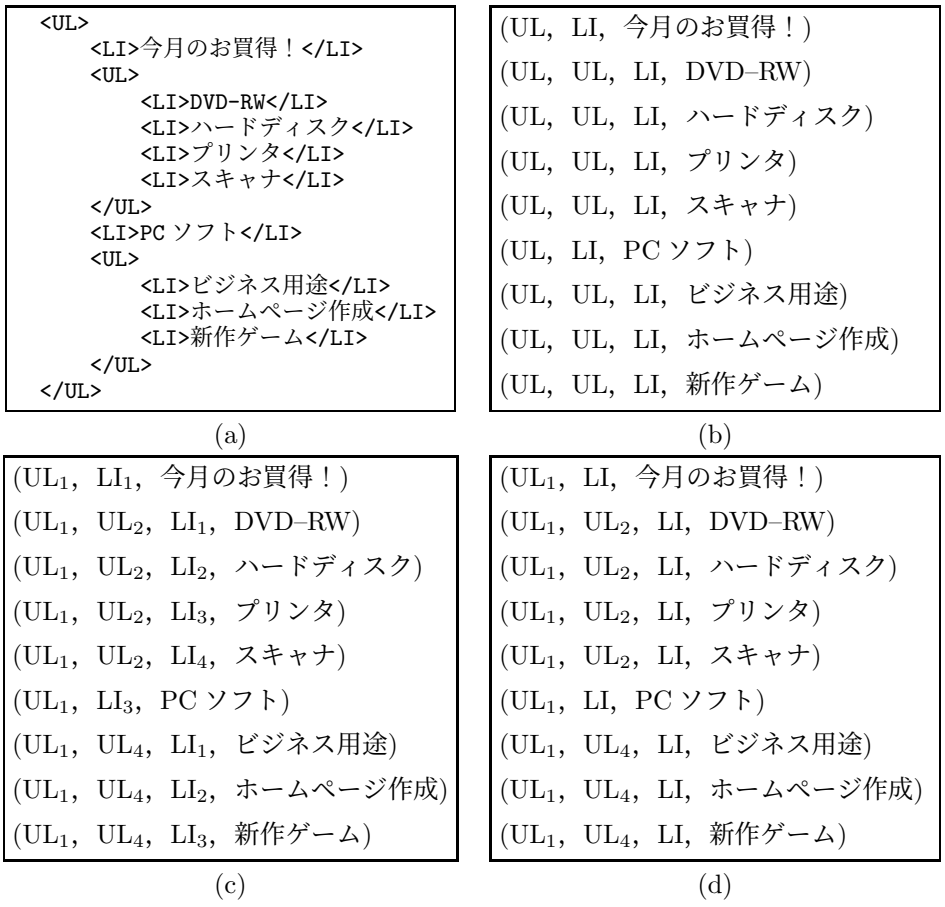

図 2　各自然言語表現のもつパス

ろうと考えられる意味的に類似した表現の集合を，それら表現を囲んでいる HTML タグに注目して獲得する．ステップ 1 は HTML 文書中のテーブル要素を転置する「前処理」，HTML 文書中のタグ情報に基づいて下位語候補集合を獲得する「下位語候補集合獲得処理」，獲得された下位語候補集合を整理する「後処理」の 3 つの処理からなる．以下では，ステップ 1 において最も重要な下位語候補集合獲得処理について説明する．残りの前処理及び後処理については付録を参照されたい．

要素がリストボックスの要素であることを意味する<OPTION>タグ，要素が箇条書きの要素であることを意味する<LI>タグ，文字を強調する<STRONG>タグなどを含む（より正確にはそれらのタグをパスの最後に含む）繰り返し要素が抽出されている．

下位語候補集合獲得処理

以下では図 2(a) に示した HTML 文書のソースコード[4]を例に，下位語候補集合の獲得方法について述べる．下位語候補集合を獲得するにあたり，HTML 文書中で繰り返し現れている要素を認識する必要がある．そのため本研究では，まず HTML 文書中に現れる各表現の**パス**を求める．ここでいうパスとは，HTML 文書中の表現がどのようにタグ付けされているかを表すものであり，表現を囲んでいるタグをそのネストの順序にしたがって，リスト形式で表したものである．図 2(a) において，表現「今月のお買得！」はタグ<LI>，</LI>に囲まれており，さらに<UL>，</UL>にも囲まれている．これらのタグを，表現「今月のお買得！」を囲む順序にしたがって並べれば，そのパスとして (UL，LI，今月のお買得！) が得られる．図 2(a) に示した HTML 文書中の各表現は図 2(b) のようなパスを持っている．

下位語候補集合獲得処理では，HTML 文書中に現れる同じパスを持つ表現同士をまとめ，下位語候補集合として獲得する．しかし，ただ単に同じパスを持つ表現を集めてきただけては意味的に類似した下位語候補集合を獲得することはできない．例えば図 2(b) の場合，同じパスを持つ表現同士をまとめると，

{DVD–RW，ハードディスク，プリンタ，スキャナ，ビジネス用途，ホームページ作成，
新作ゲーム }
{ 今月のお買得 !，PC ソフト }

という 2 つの下位語候補集合が得られるが，周辺機器と PC ソフトのジャンルが混ざっていたり，関係のない表現同士であったりと，どちらの集合にも意味的な類似性をみることができない．この原因は同一タグの出現順序を区別できていないからである．そこで，タグにその出現順序を考慮し，改めてパスを求めることにする（図 2(c)）．ここでタグ名の横の数字はタグの出現順序を表している．しかし，今度はどのパスも一意になってしまい，同じパスを持つ表現を得ることができなくなる．そこで，表現からみて N 個前のタグまでは，タグの出現順序を考慮しないようにする．図 2(c) の場合，$N = 1$ とすると図 2(d) のようなパスを得ることができる．これらを同じパスを持つ表現ごとにまとめると，

{DVD–RW，ハードディスク，プリンタ，スキャナ }
{ ビジネス用途，ホームページ作成，新作ゲーム }
{ 今月のお買得 !，PC ソフト }

というように，意味的に類似した共通の上位語を持つであろう表現の集合を得ることが可能になる．本研究では経験的に $N = 1$ として下位語候補集合の獲得を行っている．

4 図 1に示した HTML 文書のソースコードである．

3.3 *df*，*idf* に基づく上位語候補の獲得 (ステップ 2)

ステップ 1 では，HTML タグに基づいて共通の上位語を持つであろう下位語候補の集合を獲得した．ステップ 2 ではステップ 1 で獲得した各下位語候補を含む文書中から，前述した 2 番目の仮説「**共通の上位語をもつ下位語の集合が与えられた時，各下位語に共通する上位語は各下位語を（少なくとも 1 つ）含む文書に現れやすく，それ以外の文書には比較的現れにくい**」に基づき，情報検索の分野などで従来より用いられている *df* や *idf* といった統計量を利用して各下位語候補に共通する上位語候補を獲得する．

ステップ 2 では上位語候補の獲得を行うにあたり，まず 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 文書集合の中から無作為に選んだ HTML 文書からなるもので，これを**大域的文書集合**と呼ぶ．この文書集合は一般的な文脈においての単語の文書頻度を求める際に使用する．次いで 2 つ目の文書集合は，ステップ 1 で獲得された下位語候補集合の各要素を 1 つでも含む文書を，既存のサーチエンジンより収集し作成するもので，**局所的文書集合**と呼ぶ．この文書集合は与えられた下位語候補集合の各要素と，ステップ 2 で獲得する上位語候補の関連の強さを測る際に用いる．

以下では，ステップ 1 より獲得された下位語候補集合を C，大域的文書集合を G，C の各要素を検索語として収集した局所的文書集合を $LD(C)$ と記述する．また，$LD(C)$ に含まれる全ての名詞の中から，普通名詞，サ変名詞，地名を表す名詞を抽出し，その中から上位語としては好ましくない語[5]を削除して得られる名詞の集合を N とする．ステップ 2 では，上位語候補 $h(C)$ を以下の式により求める．

$$h(C) = argmax_{n \in N}\{df(n, LD(C)) \cdot idf(n, G)\}$$

$$idf(n, G) = \log \frac{|G|}{df(n, G)}$$

ここで $df(n, D)$ は，文書集合 D 中で名詞 n を含む文書数を返す関数であり，$|G|$ は文書集合 G に含まれる文書数を表す．上式は，局所的文書集合中の多くの文書に現れ，かつ大域的文書集合中の文書には相対的にあまり現れない名詞を上位語候補として獲得する．

また，下位語候補集合中の特定の要素のみに関連の強い語，例えば 3 節冒頭で挙げた DVD–RW，ハードディスク，プリンタ，スキャナからなる下位語候補集合の例で言えば，プリンタに対する「インク」という語は，この時点で上位語候補として獲得されにくい．その理由は，全下位語候補を検索語として得られた文書集合中での各名詞の文書頻度を上式では用いているため，特定の下位語候補にのみ関連の強い語の文書頻度は，下位語候補全体に関連の強い語の文書頻度より低くなり，$df(n, LD(C)) \cdot idf(n, G)$ で求められるスコアも相対的に低くなりやすいからである．

5 これら不要語の詳細については，付録 B を参照のこと

また本研究では，上位語候補を獲得する際，各名詞のスコア付けに $df(n, D)$ を用いているが，これを文書集合 D 中における名詞 n の出現頻度を求める関数 $tf(n, D)$ に変更することも可能である．$tf(n, D)$ に変更すると，従来より単語の重みを計算する際に利用されている $tf{\cdot}idf$ 法と同じになる．本研究では，各名詞のスコア付けに $tf(n, LD(C)) \cdot idf(n, G)$ を用いた場合についても上位語獲得実験を行った．しかし，$df(n, LD(C)) \cdot idf(n, G)$ を用いた場合と比べ，高い精度で正しい上位下位関係を獲得するには至らなかった．詳細については 4 節にて述べる．

3.4 意味的類似度に基づく上位語候補と下位語候補集合の並べ替え (ステップ 3)

ステップ 2 までで下位語候補集合とその上位語候補の組を獲得することができた．これらの組の集合を以下の形式で表すことにする．

$$\{\langle h(C_1), C_1\rangle, \langle h(C_2), C_2\rangle, \cdots, \langle h(C_m), C_m\rangle\}$$

ここで $C_1, \cdots, C_m$ はステップ 1 より獲得された下位語候補集合を表しており，$h(C_i)$ はステップ 2 で獲得された C_i の各要素に共通の上位語候補を表している．ステップ 3 では，先述した 3 番目の仮説「**上位語と下位語は意味的に類似しており，その類似性は上位語と下位語の持つ係り受け関係によって捉えることができる**」に基づき，$h(C_i)$ と C_i の各要素の持つ係り受け関係から両者の意味的類似度を計算し，その類似度とステップ 2 で計算した上位語候補の持つ $df \cdot idf$ 値の両方を考慮したスコアに基づいて $\langle h(C_i), C_i\rangle$ 各組の順位付けを行う．本研究で提案する手法は，ステップ 3 で求める順位に基づき，後述するステップ 4 を適用後，その上位 k 組を最終的な上位下位関係として出力する．すなわち，残りの $m - k$ 組は，間違った上位語が獲得されやすいという理由から削除する．

3.1 節でも述べたようにように，ステップ 2 の結果には上位語ではないが各下位語候補と非常に関連の強い語を上位語候補として獲得してしまっている組が存在する．例えば，DVD–RW，ハードディスクなどからなる下位語候補集合の例に対し，その上位語候補として「機器」ではなく「データ」を獲得してしまっているケースもある．ステップ 3 では，このような誤った上位語が獲得されている上位語候補と下位語候補集合の組を，最終的な出力結果から削除する．本研究では，下位語候補集合の各要素と関連は強いが上位語ではない語（先程の例でいえば「データ」）は，下位語候補集合の各要素との意味的な類似性が弱く，逆に妥当な上位語（「機器」）は意味的な類似性が強いと考える．そのため，上位語候補と下位語候補間の意味的類似度に従って上位語候補と下位語候補集合の組を順位付けすれば，誤った上位語が獲得されている組に対して低い順位を付けることが期待できる．このように，誤ったものを低く，正しいものを高く順位付けし，その上位幾つかを最終的な獲得結果とすることで，相対的に高い精度で上位下位関係の獲得が可能になる．

ステップ 3 では上位語候補と下位語候補間の意味的類似度を計算するために，まず，局所的文書集合中から，各下位語候補が含まれる文を抽出し，係り受け解析を行う．そして係り受け解析の結果から，各下位語候補がある助詞を介してある動詞にかかる係りやすさを計算する．ここで，下位語候補集合 C の要素のいずれかが助詞 p を介して動詞 v に係る頻度を，$f_{hypo}(C,p,v)$ と記述する．そして，すべての助詞を $\{p_1,\cdots,p_l\}$，すべての動詞を $\{v_1,\cdots,v_m\}$ で表したとき，下位語候補全体の係り受け関係を表したベクトル（以降ではこの係り受け関係を表したベクトルのことを**係り受けベクトル**と呼ぶ）を以下のように定義する．

$$hypov(C) = \langle f_{hypo}(C,p_1,v_1), f_{hypo}(C,p_2,v_1), \cdots, f_{hypo}(C,p_{l-1},v_m), f_{hypo}(C,p_l,v_m)\rangle$$

ここで，下位語候補全体から係り受けベクトルを生成しているのは，下位語候補単独の係り受け関係だけでは，出現頻度が低いために的確な係り受け関係を捉えているとは考えにくいためである．続いて下位語候補集合と同様，上位語候補 $h(C)$ の係り受けベクトルを次のように定義する．

$$hyperv(h(C)) = \langle f(h(C),p_1,v_1), f(h(C),p_2,v_1), \cdots, f(h(C),p_{l-1},v_m), f(h(C),p_l,v_m)\rangle$$

ここで $f(n,p,v)$ は，新聞記事 33 年分[6]に既存の構文解析器 (金山, 鳥澤, 光石, 辻井 2000)[7]を適用することにより求めた，名詞 n が助詞 p を介して動詞 v に係る頻度を表している．この時，新聞記事中に 500 回以上現れない名詞に関しては，係り受け関係のカウントは行わなかった．そのため，このような名詞が上位語候補として獲得された組に関しては，上位語候補と下位語候補全体との意味的類似度を 0 とした．また，今回係り受けベクトルを作成するために，新聞記事より学習した係り受けデータを用いた理由は，単に大量の新聞記事が既に構文解析済みであったためである．WWW 上から大量の文書を収集し，それらから係り受け関係を計算するのは今後の課題である．

係り受けベクトルあるいは係り受け確率を使って 2 単語間の意味的類似度を求める方法は Lin(Lin 1998) や Lee(Lee 1999) でも述べられているように，Jaccard 係数など幾つかの手法が存在するが，本研究では，文書検索などに用いられているコサイン尺度 (Salton and Lesk 1968) を用いて意味的類似度を計算する．任意の下位語候補集合 C とその上位語候補 $h(C)$ の意味的類似度は以下の式で計算される．

$$sim(C,h(C)) = \frac{hypov(C)\cdot hyperv(h(C))}{|hypov(C)|\times|hyperv(h(C))|}$$

なお，Lin や Lee らが提案しているものも含め，他の方法で上位語／下位語間の意味的類似度を計算することは，今後の課題である．

6 読売新聞 1987–2001，毎日新聞 1991–1999，日経新聞 1990–1998; 計 3.01GB

7 論文 (金山他 2000) では素性構造の単一化を行っているが，本実験で用いたバージョンでは，単一化の近似だけを行っている．

ステップ 3 では，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 m 個からなる集合 $\{\langle h(C_i), C_i\rangle\}_{i=1}^{m}$ を以下の値に基づいてソートする．

$$sim(C_i, h(C_i)) \cdot df(h(C_i), LD(C_i)) \cdot idf(h(C_i), G)$$

ここでソートを行う際に，意味的な類似度だけではなく，ステップ 2 で計算された上位語候補の $df{\cdot}idf$ 値も考慮していることに注意されたい．

また上記のスコアの異なる利用方法として，1 つの下位語候補集合に対して，ステップ 2 より獲得された上位 k 個の上位語候補を上式を用いて改めて順位付けをし，そのトップを上位語候補として獲得するという方法も考えられる．本研究でも，同様の手法を実装し上位下位関係獲得の性能を評価した．しかし，上記のスコアを用いて上位語候補を改めて順位付けすることによる有意義な精度の向上は見られなかった．その詳細については 4 節で述べる．

3.5 ヒューリスティックルールを用いた上位語候補と下位語候補集合の組の修正ならびに取捨選択 (ステップ 4)

ステップ 4 では，予備実験より得た知見に基づき作成したヒューリスティックなルールを，ステップ 3 までで得られた上位語候補と下位語候補集合の組に適用することで，上位下位関係の獲得精度の改善をはかる．ステップ 4 で使用したルールは以下の 3 つである．

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を削除する

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について上位語候補が末尾に現れていない

ルール 3　獲得された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

ルール 1 では，誤って獲得された上位語候補を持つ組を削除することで精度の改善をはかる．一般に，上位語は下位語に比べより広い文脈で使われているため，下位語を含む文書より上位語を含む文書の方が WWW 上により多く存在しているはずと考えることができる．このような一般的な上位語が持っていると考えられる特性を利用したのがルール 1 である．

次いでルール 2 では，誤った上位語候補が獲得されている組を削除するのに加え，意味的な類似性の見られない下位語候補集合を持つ組についても削除することで精度の改善をはかる．しかし，このルールは獲得された上位語候補が下位語候補の部分文字列として現れない場合には適用されない．日本語において，複合名詞の主辞は主として末尾に現れる名詞であるため，

下位語候補集合の多くの要素で共通の語が末尾に現れている場合，その語は妥当な上位語である可能性が高いと考えられる．それに対し，獲得された上位語候補が下位語候補の末尾以外の場所に現れる場合，その上位語候補は妥当な上位語である可能性は低いと考えられる．また，下位語候補集合の一部の要素の末尾にだけ上位語候補が現れる場合，その下位語候補集合には意味的な共通性が見られにくく，そのような下位語候補集合は共通する上位語も持ちにくいと考えられる．そのため，このような上位語候補や下位語候補集合を持つ組は最終的な出力結果から削除する．

最後にルール 3 では，獲得された誤った上位語候補を正しい上位語に置換することで精度の改善をはかる．予備実験において，下位語候補集合の要素が地名の場合，それら地名を含む地域を指す地名が，上位語候補として獲得されているケースが頻繁に見られた．例えば，下位語候補集合が「東京」,「埼玉」,「神奈川」,「千葉」という表現からなっていた場合，妥当な上位語としては「地域」や「都道府県」などが考えられるが，ステップ 3 までで述べた方法で上位語候補を求めると,「日本」という語が上位語候補として得られる．実際に獲得された上位語候補「日本」は，本研究でたてた仮説を満足するが,「東京」や「埼玉」,「神奈川」,「千葉」に対して「日本」という語は包含関係 (*part-whole relation*) を表す語であり，上位語ではない．そこで，地名を表す表現からなる下位語候補集合に対しても正しい上位語を獲得できるようにするため，獲得された上位語候補が地名を表す語であった場合は，それを「地名」に置き換える．

以上のルールをステップ 3 までで得られた結果に対して適用することで，幾つかの上位語候補と下位語候補集合の組が削除または修正され，性能のさらなる向上が見込めることを実験により検証する．

4 評価実験

本研究では，約 4.66×10^6 件の HTML 文書（重複なし）を WWW よりダウンロードした．そして，ダウンロードした文書集合の中から 1.00×10^6 件の HTML 文書 (約 1.26GB，タグなし) を無作為に選びだし，新しく文書集合を作成した．作成した文書集合は，一般的な文脈においての単語の文書頻度を求めるために用いる大域的文書集合として上位語候補を獲得する際に用いる．大域的文書集合から単語の一般的な文書頻度を求めるにあたり，大域的文書集合中に含まれる各 HTML 文書からタグを除去し，JUMAN(黒橋 長尾 1999) を用いて形態素解析を行った．そして，そこに現れる単語の文書頻度を計算し，文書頻度のリストをを作成した．この時，文書頻度が 30 に満たない単語に関しては，的確な文書頻度が得られていないと考え，リストから除外している．

次いで，評価実験に用いる下位語候補集合を獲得するため，先程の 4.66×10^6 件の文書集合より，約 8.71×10^5 件の HTML 文書 (約 0.89GB，タグなし) を選びだした．そして，HTML 文書中で省略されている終了タグの補完や，誤ったタグの入れ子構造を持つ部分を適切な入れ

表 2　実際に獲得された下位語候補集合の例

ID	下位語候補集合の要素
10397	広井法代, 山川純子, 池田和子, 柏木久美子
16653	わからない, 回だけ, 回以上, 回以上回未満
21561	あじさい茶屋, まぐろ市場, クローバー, ドトール, ドムドム, ベックス, ラガール, ランパデール, 小竹林, 東神奈川そば店, 道中そば, 本郷台そば店
28931	DDI ポケット, ドコモ関西, ドコモ四国, ドコモ東海, ドコモ北陸, 九州通信ネットワーク, 四国情報通信ネットワーク
30288	違法でない, 違法又は不当, 勧告, 義務を果たしているか, 通知, 日以内, 理由がある, 理由がない
35645	阿部委員, 紫芝委員, 津金委員, 眞柄委員
51462	旧海運局, 旧陸運局, 工事発注見通しの公表, 所在地, 情報公開
53147	なし, 円, 朝回, 不可
56681	NTT 東日本, アイコム, オムロン, コレガ, プラネックス, マイクロ総研, メルコ, ヤマハ
58174	監督, 原作, 作品A, 時間, 出演
59502	さんた, ふるみそ, ほぴの, やまおり, やまそと
69064	学会, 技術, 研究, 工学, 通信, 電子, 福祉

子構造への変換を行う，フリーのユーティリティである HTML Tidy[8]を用いて，各 HTML 文書を XML 文書へ変換し，3.2 節で述べた手法により下位語候補集合の獲得を行った．その結果，9.02×10^4 個の下位語候補集合（重複あり，全部で 6.01×10^5 個の下位語候補を含んでいる）を獲得した．実際に獲得された下位語候補集合の例を表 2に示す．続いて，9.02×10^4 個の下位語候補集合の中から重複を除き，無作為に選択した 2,000 個を評価実験に用いるテストセットとした（本手法の開発には，テストセットとは重複しない約 4,000 個の下位語候補集合を用いている）．このテストセットとして選択した 2,000 個の下位語候補集合には，全部で 13,790 個の下位語候補が含まれている．

次に，3.3 節で述べた方法で上位語候補を獲得するにあたり，個々の下位語候補を検索語として検索エンジン goo[9]より検索し，その結果得られた文書集合のうち上位 100 件をそれぞれダウンロードして局所的文書集合を作成した．この時，検索結果が 100 件に満たない下位語候補に関しては，検索により得られた全ての文書をダウンロードした．作成した局所的文書集合は，大域的文書集合と同様，その中に含まれる各 HTML 文書からタグを除去し，JUMAN を用いて形態素解析を行った．そして，下位語候補全体の係り受けベクトルを求めるため，既存の構文解析器 (金山他 2000) を用い係り受け解析を行った．

本研究では，いくつかの異なる実験を行っているが，各実験において上位下位関係獲得の精度を次のようにして求め，グラフを作成している．まず，ソートされた上位語候補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，その中で正しい上位下位関係が獲得されている割合を計算し，それを上位下位関係獲得の精度とした．グラフの横軸は，ソートされた上

8 http://www.w3.org/People/Raggett/tidy/
9 http://www.goo.ne.jp/

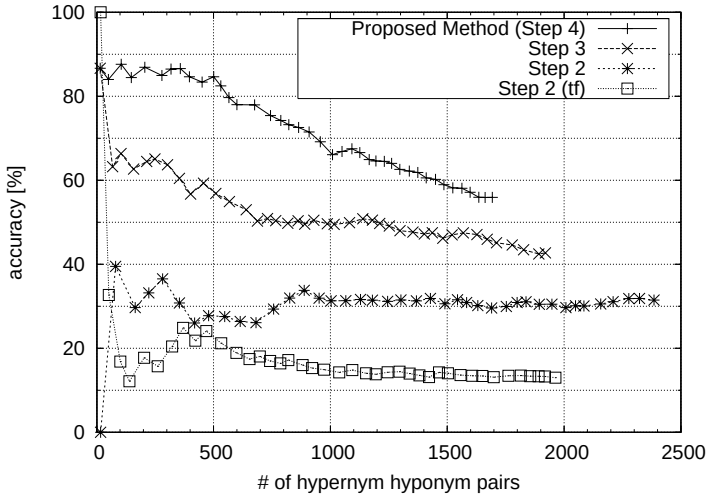

図 3　各ステップを経ることでの精度の移り変わり

位語候補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，縦軸は正しい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．

$$\langle \sum_{k=1}^{j} |C_k|, \frac{\sum_{k=1}^{j} correct(C_k, h(C_k))}{\sum_{k=1}^{j} |C_k|} \rangle$$

ここで j は $1 \leq j \leq 200$ であり，$|C_k|$ は下位語候補集合 C_k の要素数である．さらに $correct(C_k, h(C_k))$ は実際に獲得された上位語候補 $h(C_k)$ が正しい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．

最初の評価実験として，ステップ 2，3，4 を経ることにより上位下位関係の獲得精度が変化することを確認するため，各ステップごとに獲得された上位下位関係の評価を行った．図 3 に，ステップ 2，3，4 を経ることによって変化する精度の様子を示す．各ステップでは，獲得された上位語候補と下位語候補集合の組のソートを行っている．ソートの基準として，ステップ 3，4 では獲得された上位語候補の $sim(C, h(C)) \cdot df(h(C), LD(C)) \cdot idf(h(C), G)$ のスコアを，ステップ 2 では $df(h(C), LD(C)) \cdot idf(h(C), G)$ のスコアを用いている．つまり，ステップ 2 までで獲得された結果をソートする時は，上位語候補と下位語候補集合の類似度は考慮されていない．また，各ステップでは上位語候補と下位語候補集合の組のソートを行った後，全体の 1 割にあたる上位 200 組を最終的に獲得された上位下位関係として評価対象にしている．残りの 1,800$(= 2,000 - 200)$ 組は，前述したように間違った上位語が獲得されやすいという観点から評価対象から外した．図 3中で “Step 4” と示した曲線が，今回提案した手法により最終的に

獲得された上位下位関係獲得の精度を表している．この図より獲得された全上位下位関係数の約 3.6%にあたる上位 501 個（獲得された上位語の異なり数は 36 個）の関係を取り出した場合，その精度はおよそ 84.6%を示している．さらに，全関係数の約 5%にあたる上位 701 個（獲得された上位語の異なり数は 44 個）の関係を取り出した場合，その精度はおよそ 75%，約 10%にあたる上位 1398 個（獲得された上位語の異なり数は 93 個）の場合で 61%と，獲得する上位下位関係数を多くするほど徐々にその精度が落ちていくことが図よりわかる．このことから，$sim(C, h(C)) \cdot df(h(C), LD(C)) \cdot idf(h(C), G)$ のスコアに基づいて，上位語候補と下位語候補集合の組をソートすることで，正しい上位語が獲得されている組に対しては高い順位を，誤った上位語が獲得されている組に対しては低い順位をつけることができていることがわかる．

3.3 節で述べたように，ステップ 2 において $df(n, LD(C)) \cdot idf(n, G)$ ではなく，$tf(n, LD(C)) \cdot idf(n, G)$ により各名詞のスコア付けを行い上位語候補を獲得した結果が図 3中の “Step 2(tf)” で示したグラフである．図より，$tf(n, LD(C)) \cdot idf(n, G)$ に比べて $df(n, LD(C)) \cdot idf(n, G)$ の方が上位語候補獲得の精度が高いことが確認できる．この結果から上位語候補獲得というタスクにおいては，$tf(n, LD(C)) \cdot idf(n, G)$ より $df(n, LD(C)) \cdot idf(n, G)$ の方が適していることがわかる．

表 3に実際に獲得された上位語と下位語候補集合の例を示す．下位語候補集合の幾つかの要素がその末尾に共通の語を持っている場合，その語が上位語になりやすいというのは自明であるため，表 3にそのような例は載せていない．また，この表はステップ 4 で獲得された最終的な結果だけでなく，ステップ 3 までで獲得された上位語候補と下位語候補集合の組の結果も示している．そのため，ステップ 4 の各ルールを適用することで，ステップ 3 で獲得された結果のうち幾つかが修正または削除される様子が確認できる．

次に各ステップ，およびステップ 4 で用いている各ルールがどのくらい精度の向上につながっているのかを確認するため，各ステップ，およびステップ 4 中の各ルールを適用しなかった時での上位下位関係獲得の性能を評価した．その結果を図 4に示す．図 4において “– Step X” もしくは “– Rule X” となっているものは，今回提案した手法から “ステップ X”，または “ルール X” を抜いた時の精度を表している．図 4より，どのステップやルールも精度の向上につながっていることが確認できる．上位 200 組の上位語候補と下位語候補集合の組を獲得した時，ステップ 3 を抜いた場合で約 7.7 %，ステップ 4 を抜いた場合では約 13.2 %の精度の低下が見られる．このことから，ステップ 3 よりもヒューリスティックなルールを適用することで精度の向上を図るステップ 4 の方が，精度の向上により働いていることがわかる．また，ステップ 4 の中でもとりわけ，地名を表す上位語を「地名」に変換するルール 3 を抜いた場合が最も精度が落ちていることから，提案手法により獲得された上位下位関係の中には地名に関するものが多く含まれていると考えられる．実際，提案手法により獲得された正しい上位下位関係の数は 947 個であるが，ルール 3 を抜いた場合では 779 個に減少しており，この数字から正しく獲得

表 3　提案手法により獲得された下位語集合とその上位語の例

Step4 での順位	Step1 で獲得された下位語候補集合	Step2 で獲得された上位語	Step3 での順位	Step4 のルール			Step4 で獲得された上位語
				1	2	3	
10	朗読者*, オブジェクト指向入門*, 月の砂漠をさばさばと*, もこもこもこ*, ソフトウェア職人気質*, 入門 JavaScript*	本	23	–	–	–	本
16	テディベア*, チョウカイリョウガ*, ヨシフサキング*, プラントタイヨオー*, ナスノホシヒメ*, フローレスライン*, ノーザンカピタン*, ミヤビリージェント*, クラレットパンチ*, トーセンダンディ*, アーサーズフェイム*, ケイアイチャンス*, ロイスジュニア*, カナハラドラゴン*, ウインシュナイト*, ダイワサイレンス*, マチカネラッパ* , マイネルグリズリー*, ミスタードン*	馬	42	–	–	–	馬
21	コオリガモ*, ビロードキンクロ*, アカハジロ*, クビワキンクロ*, メジロガモ*, アカハシハジロ*, キンクロハジロ*, コケワタガモ*, スズガモ*, ホオジロガモ*, シノリガモ*, クロガモ*, ホシハジロ*, ケワタガモ*, ヒメハジロ*, アラナミキンクロ*, オオホシハジロ	鳥	53	–	–	–	鳥
29	殺人*, 放火*, 強姦*, 侵入盗*, 侵入強盗*, 非侵入盗*, 非侵入強盗*	犯罪	68	–	–	–	犯罪
47	将軍*, 宮本武蔵*, 羅生門*, 七人の侍*, ミッドウェイ*, 無法松の一生*, 太平洋の地獄*, 武士道ブレード*, 価値ある男*, 用心棒』『赤ひげ, 大統領の堕ちた日*	映画	112	–	–	–	映画
69	モスクワ*, キエフ*, タシケント*, ミンスク*, トビリシ*, ドゥシャンベ*, ビシュケク*, アスタナ*, キシニョフ*, エレバン*, バクー*, アシハバード*	ロシア	169	–	–	+	地名
78	福留宏紀*, セギノール*, 藤井康雄*, シェルドン*, 五島裕二*, 玉木朋孝*, 塩谷和彦*, 平野恵一*,	選手	196	–	–	–	選手
81	ワイヤレスカード, 小電力セキュリティ, ＰＨＳ陸上移動局, 市民ラジオ, 特定小電力機器,	無線	200	–	–	–	無線
82	大切なもの*, もらい泣き*, 大きな古時計*, 星屑の街*, 白い花*, 未完成のメロディ*	曲	201	–	–	–	曲
86	踊る大捜査線*, プロジェクトＸ, 世紀を越えて, 彼女たちの時代*	ドラマ	207	–	–	–	ドラマ
106	桑田真澄*, 上原浩治*, ワズディン*, 武田一浩*, 木村龍治*, 真田裕貴*, 鄭ミン台*, 趙成ミン*	投手	250	–	–	–	投手
116	イワウメ*, チシマザサ, キバナシャクナゲ*, ミヤマナルコユリ*	花	280	–	–	–	花
127	シイタケ*, サンゴハリタケ*, サンコタケ*, シロオニタケ*, シロイボカサタケ*	キノコ	306	–	–	–	キノコ
139	音楽, 映画, マンガ, 出会い, 芸能人	サイト	324	–	–	–	サイト
150	夏目漱石, 芥川竜之介, 鷹野つぎ, 国木田独歩, 徳冨蘆花, 菊池寛, 若山牧水, 梶井基次郎, 夢野久作, 宮本百合子, 田中貢太郎, 夢野久作海若藍平 ブレイクウィリアム,	作品	343	–	–	–	作品
172	新年, 万聖節, 主顕節, メーデー, クリスマス, イースター, 解放記念日, 聖母受胎祭, 聖ステファノの日, 聖母昇天祭	日本	391	–	–	+	地名
–	銀河群, 構成メンバー, 局部銀河群 アンドロメダ銀河*, 銀河系*,	銀河	10	–	+	–	–
–	ブラジル, フィリピン, 韓国, インド, アメリカ, タイ 中国, ペルー, オーストラリア, アルゼンチン, スペイン	日本	80	+	–	+	–

“*” が後についている下位語候補は，提案手法により妥当な上位語が獲得されたものを示す．また，ルールに対応した列にある ‘+’ は，対応するルールが実際に適用され，上位語候補ならびに下位語候補集合の対が削除された，あるいは修正されたことを示す．

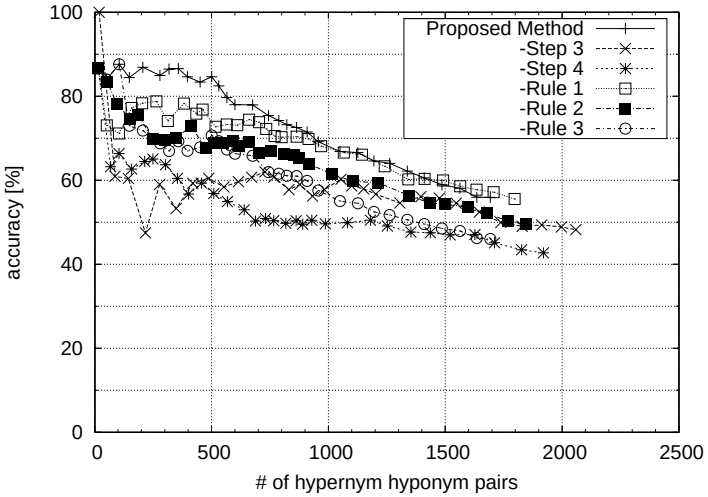

図 4　各ステップ及び各ルールの効果

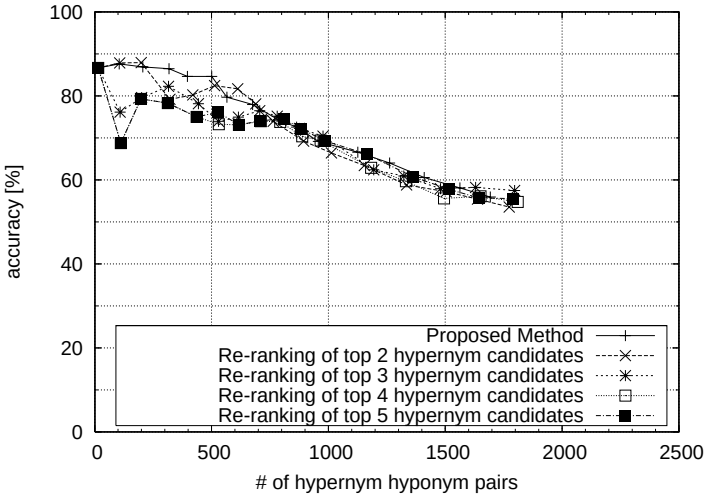

図 5　リランキングの効果

された上位下位関係のうち，およそ 17.7％は HYPONYM(“地名”, “東京”) のような上位下位関係で占められていることがわかる．

最後に 3.4 節で触れたように，ステップ 2 より獲得された上位語候補上位 k 個を $sim(C, h(C)) \cdot df(h(C), LD(C)) \cdot idf(h(C), G)$ のスコアで改めて順位付けを行い，そのトップを上位語候補として獲得した場合の，上位下位関係獲得実験の結果を図 5に示す．図より，

$sim(C, h(C)) \cdot df(h(C), LD(C)) \cdot idf(h(C), G)$ のスコアを用いて上位語候補をリランキングしても，有意義な精度の向上がみられないのがわかる．

5 他の手法との比較実験

評価実験により，本研究でたてた 3 つの仮説，ならびに 3 つのヒューリスティックルールが単語間の上位下位関係の獲得に有効であることがわかった．今度は，その他の手法と比べて良いかどうかの比較を行う必要がある．そこで，本節では提案手法とその他の手法との比較実験を行う．今回，比較対象の手法として以下に示す 4 つ手法を挙げた．

手法 1 複数の下位語候補の末尾で共有される語を上位語として獲得する方法を手法 1 とする．これは，日本語が head-final な言語であるため，下位語候補集合中の多くの要素の末尾に共通して現れるような語は，妥当な上位語である可能性が高いという考えに基づいている．手法 1 ではこの考えに基づき，まず複数の下位語候補の末尾に共通して現れる語を収集する．そして，収集された語の中で最も文字列長の長い語を上位語として獲得し，それが妥当な上位語であるかどうかを評価する．

手法 2 下位語候補集合を獲得した HTML 文書中の箇条書きや表データのキャプションから，上位語を獲得する方法を手法 2 とする．一般に HTML 文書中に現れる箇条書きや表データのキャプションには，上位語が含まれやすいと考えられる．そこで手法 2 では，下位語候補集合を獲得した箇条書きや表データの直上，もしくはさらにその 1 つ上に存在するキャプションを人手で抜き出し，その中に適切な上位語が含まれているかどうかを評価する．少なくとも現段階では HTML 文書中の箇条書きや表データのキャプションから自動的に上位語を獲得する手法がないため，この手法により得られた上位下位関係獲得の精度は，キャプションから上位語の獲得を試みる手法の上限だということに注意されたい．

手法 3 今角 (今角 2001)，安藤ら (安藤他 2003) の研究で用いられている構文パターンを基に正規表現パターンを作成し，それにより上位語を獲得する方法を手法 3 とする．手法 3 で用いている正規表現パターンを表 4に示す．手法 3 では，提案手法によりあらかじめ獲得された正しい上位下位関係を正規表現パターンに与え，そのパターンに適合する文が与えられた文書集合中に現れるかどうかを評価の対象としており，先に挙げた手法 1 及び手法 2 とはその評価基準が異なる．つまり手法 3 では，本研究で提案した手法により獲得された正しい上位語と下位語の組を，正規表現パターンを用いて獲得できるかどうかだけしか確認していない．また，手法 3 で用いているパターンは，構文解析結果ではなく文の表層と正規表現のマッチングだけを利用して上位下位関係の獲得を行っているため，先行研究で用いられている構文パターンとは若干異なる．この若干の差異が，

表 4　比較実験に用いた正規表現パターン

先行研究で用いられている構文パターン	本研究で実装した定型パターン
名詞句 A「**名詞句 B**」	**上位語**「**下位語**」
名詞句 A に似た**名詞句 B**	**下位語** .* に似た .* **上位語**
名詞句 A と呼ばれる**名詞句 B**	**下位語** .* と呼ばれる .* **上位語**
名詞句 A 以外の**名詞句 B**	**下位語** .* 以外の .* **上位語**
名詞句 A のような**名詞句 B**	**下位語** .* のような .* **上位語**
名詞句 A という**名詞句 B**	**下位語** .* と (い | 言) う .* **上位語**
名詞句 A など**名詞句 B**	**下位語** .* など (, | の)? .* **上位語**
名詞句 A などの**名詞句 B**	
名詞句 A など, **名詞句 B**	
	下位語 .* (ら | たち) .* **上位語**

各上位語と下位語は「」や “” で囲まれていても構わない.

何らかのエラーの原因となるとおもわれるかもしれないが，少なくともこの手法で使われている正規表現パターンは，先行研究で用いられている構文パターンで獲得される上位下位関係を漏れなく獲得することができるため，今回の評価基準の場合は問題ないと思われる．(逆にいうと，今回の実験結果での精度は本来の構文解析結果を用いる方法にくらべると若干高くなっているはずであり，正規表現パターンを用いることにより得られる上位下位関係獲得の精度が，構文パターンを用いて上位下位関係の獲得を試みる手法の上限であると言える．) また，手法 3 では正規表現パターンを適用する文書として，ステップ 2 で上位語候補を獲得する際に用いた局所的文書集合中に含まれている HTML 文書からタグを除いたものを利用している．そのため，もし提案手法の方が手法 3 よりも良い結果が得られれば，少なくとも，少量のテキスト，即ち，下位語候補 1 つあたり最大で 100 文書を利用する場合には，構文パターンにより獲得することができない上位下位関係を，提案手法は同量の文書から獲得できるということになる．

手法 4　手法 1，2，3 を組み合わせたものを手法 4 とする．その評価は，本研究で提案した手法で獲得できた正しい上位下位関係のうちどのくらいの関係を手法 4 で獲得できるか，という観点で行った．この時，正しい上位下位関係が獲得できているかどうかの判定は，今回提案した手法により獲得された正しい上位語と下位語の組を，手法 1，2，3 のいずれかで獲得できていれば，正しい上位下位関係が獲得できたとした．比較実験により得られる提案手法と手法 4 の精度の差は，提案手法で獲得できて，手法 1，2，3 では獲得できない正しい上位下位関係数の差を表す.

図 6に提案手法と手法 1，2，3，4 の上位下位関係獲得精度を示す．図 6では，各グラフとも提案手法と同じ方法で下位語候補集合をソートしており（つまり手法 1，2，3，4 とも，$sim(C, h(C)) \cdot df(h(C), LD(C)) \cdot idf(h(C), G)$ のスコアを用いてソートしている），その上位語を獲得する方法だけが異なっている．この図より，今回比較対象として挙げた 4 つの手法と

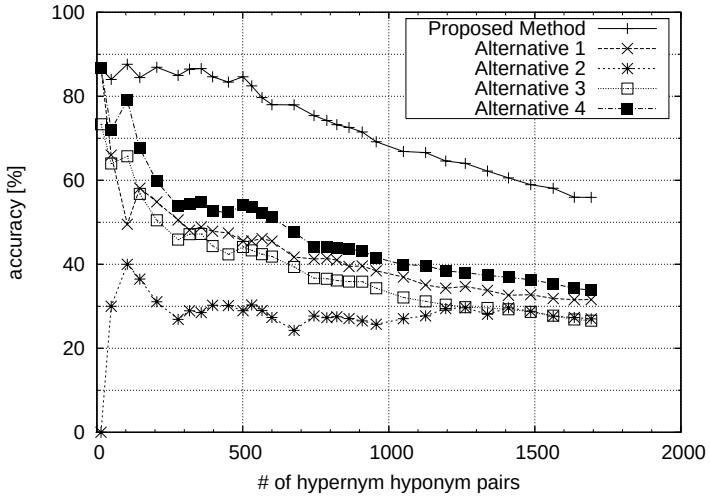

図 6　提案手法と他の手法の比較

比べ，本研究で提案した手法はより多くの上位下位関係を獲得できていることがわかる．このことから，1 つの下位語候補あたり最大で 100 件の文書を収集し作成した文書集合から上位下位関係を獲得する場合においては，手法 1，2，3，4 では獲得できないような関係を，提案手法はかなりの数獲得できているということがわかる．

手法 1 と提案手法を比べると，両者の獲得精度の間に有意義な差が見られることから，下位語候補の字面を見るだけでは獲得できないような上位語を提案手法により獲得できていると考えられる．このことから，ステップ 2 で述べた統計的な尺度（$df(n, LD(C)) \cdot idf(n, G)$）を用いて，HTML 文書集合の中から上位語候補を発見することの有効性が示されたと考えられる．

次に，下位語候補集合を獲得した HTML 文書中の箇条書きや表データのキャプションから上位語を獲得する方法である手法 2 のグラフを見ると，その獲得精度はおよそ 30 %程度であることが確認できる．HTML 文書中の箇条書きや表データから獲得した下位語候補集合に対する上位語の獲得を考えた場合，安直に下位語候補集合を獲得した箇条書きや表データのキャプション中に含まれる語を対象にして上位語の獲得を行えば良いと考えるかもしれない．しかし，実際にそれらのキャプションから上位語の獲得を試みるとその精度は最高でも 30 %程度でしかなく，高い精度で正しい上位語を獲得するこが難しいということがこのグラフからわかる．さらに，今回の場合は計算機を用いて自動的に箇条書きや表データのキャプションから上位語を獲得しているわけではなく，人手によって獲得しているため，実際に計算機を用いて箇条書きや表データのキャプションから上位語の獲得を行った場合は，さらに精度が下がると予想される．

最後に各手法と提案手法の上位下位関係獲得精度の差，特に手法 3 との差というのは，大量の文書を WWW より集め，それをコーパスとして用いることで縮まる可能性があると考えられる．しかし，大量の HTML 文書を用いて比較実験を行うことは，文書をダウンロードするのに多大な時間を要するため難しい問題であり，そのような条件下で比較実験を行うことは今後の課題である．

6　まとめと今後の課題

本稿では，構文パターンを用いずに WWW 上の HTML 文書から単語間の上位下位関係を獲得する手法を提案し，実験によりその有効性を示した．具体的には，HTML タグにより与えられる HTML 文書の構造，情報検索の分野などで用いられる *df*，*idf* などの統計量，新聞記事より獲得した名詞と動詞の係り受け関係，予備実験より得た知見に基づき作成したヒューリスティックなルールの 4 種類の要素を用いることで，少くとも，利用可能な文書の量が比較的少い場合には既存の方法では獲得できないような上位下位関係であっても，今回提案した手法で同量の文書から獲得できることが実験により確認できた．別の言い方をすれば，少なくとも少量の文書から上位下位関係を獲得する際に今回提案した手法が有効であることがわかった．ただ，残念ながら，文書の量をより増やしたときの精度の比較に関しては，今後の研究あるいは実験を待つ必要がある．

次に今後の課題について述べる．今回提案した手法により上位下位関係の獲得を試みると，その精度は獲得する上位下位関係数が増えるに従って低下する．例えば，獲得された下位語候補集合の上位 1 割を最終的な出力とした場合，その獲得精度は 60%程度であり，十分高い精度で上位下位関係の獲得が行えているとは言えない．そのため，今後の課題として，まず上位下位関係の獲得精度の向上が挙げられる．幸い，今回提案した手法は，構文パターンを用いて上位下位関係を獲得する既存の手法と組み合わせることが可能であり，それにより精度の向上が見込めるのではないかと考えている．

また，現在の方法の問題点は，実際に自然言語処理アプリケーション（例えば対話システムや情報検索システムなど）中で上位下位関係を利用するプログラマーの要求する上位下位関係を獲得できないということである．ここでプログラマーが要求する上位下位関係とは，プログラマーがアプリケーション内で使用したい上位語と，その上位語の下位語からなる上位語／下位語の対を指す．実際にアプリケーションを開発するにあたっては，どういった単語間の上位下位関係が欲しいのかということが予め決まっていることが多い．例えば，道案内を行う対話システムにおいて上位下位関係を利用することを考えた場合,「ジョナサン」や「ガスト」などからなる下位語候補集合に対しては，上位語として「店」が獲得されるよりも「ファミレス」や「レストラン」などが獲得された方が，よりきめの細かい処理ができるという点で望ましい場合が多い．そのため，プログラマーの要求する上位下位関係を自動獲得できるように本提案手法

を拡張することが今後の課題の 1 つであると考えている．

また 3 つ目の課題として，複数の語からなる上位語の獲得を行いたいと考えている．現在の提案手法では，DVD–RW，ハードディスク，プリンタなどのパソコン周辺機器からなる下位語候補集合に対しては「機器」としか上位語を求めることができないが，なんらかの手法により「周辺機器」や「パソコン周辺機器」という上位語が獲得できれば，他の自然言語アプリケーションにとって，より有用な情報となるのではないかと考えている．

謝辞 本研究を進めるにあたり，文部科学省科学研究費補助金 (平成 15 年度若手研究 (A)15680005, 平成 15 年度萌芽研究 15650015) ならびに文部科学省科学技術振興調整費 (任期付若手研究員支援プログラム、新興分野人材養成プログラム) の支援を受けた．記して謝意を表する．

参考文献

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

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

Hearst, M. A. (1992). “Automatic acquistition of hyponyms from large text corpora.” In *Proceedings of the 14th International Conference on Computational Linguistics*, pp. 539–545.

Lee, L. (1999). “Measures of Distributional Similarity.” In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, pp. 25–32.

Lin, D. (1998). “Automatic Retrieval and Clustering of Similar Words.” In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, pp. 768–774.

Mandala, R., Tokunaga, T., and Tanaka, H. (1998). “The Use of WordNet in Information Retrieval.” In *Proceedings of the COLING–ACL workshop on Usage of Wordnet in Natural Language Processing*, pp. 31–37.

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

Morin, E. and Jacquemin, C. (2003). “Automatic acquisition and expansion of hypernym links.” In *Computer and the Humanities 2003*. forthcoming.

Salton, G. and Lesk, M. E. (1968). “Computer evaluation of indexing and text processing.” *Journal of the ACM*, **15** (1), 8–36.

Yoshida, M., Torisawa, K., and Tsujii, J. (2001). “A method to integrate tables of the World Wide Web.” In *Proceedings of the International Workshop on Web Document Analysis*, pp. 31–34.

安藤まや, 関根聡, 石崎俊 (2003). “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.

今角恭祐 (2001). “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.” Master’s thesis, 九州工業大学.

岡本潤 石崎俊 (2001). “概念間距離の定式化と電子化辞書との比較.” 自然言語処理, **8** (4), 37–54.

黒橋禎夫 長尾真 (1999). 日本語形態素解析システム JUMAN version 3.61 使用説明書. http://www.kc.t.u-tokyo.ac.jp/nl-resource/juman.html.

金山博, 鳥澤健太郎, 光石豊, 辻井潤一 (2000). “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, **7** (5), 71–91.

略歴

新里 圭司: 2002 年東京電機大学工学部情報通信工学科卒業．2004 年北陸先端科学技術大学院大学情報科学研究科 博士前期課程修了，現在 同大学院博士後期課程に在学中．また，2004 年より文部科学省科学技術振興調整費研究員．新聞記事／ HTML 文書を対象とした単語間の関係の自動抽出及び，抽出した単語間の関係を基にした辞書の自動構築・拡張の研究に興味を持つ．

鳥澤 健太郎: 1992 年東京大学理学部情報科学科卒業. 1995 年同大学大学院理学系研究科情報科学専攻博士課程退学, 同年より同大学院理学系研究科情報科学専攻助手. 1998 年より 2001 年まで科学技術振興事業団さきがけ研究 21 研究員兼任. 2001 年より北陸先端科学技術大学院大学情報科学研究科助教授. 計算言語学の研究に従事. 博士（理学).

(年　月　日 受付)
(年　月　日 再受付)
(年　月　日 採録)

付録 A 下位語候補集合を獲得する際の前処理および後処理

以下では，3.2 節で説明した下位語候補集合の獲得（ステップ 1）において，下位語候補集合獲得処理の前後で行っている前処理及び後処理について述べる．

ふりがな	詳細	サーチエンジン	備考
終わりに	終りに	電話番号	コメント
おわりに			
^トップ	^ホーム	^リンク	^ヘルプ
^ニュース	^プレゼント	^カテゴリ	^サポート
^お問い合	^次の	^前の	^新着
^メール			
履歴$	リンク集$	連絡先$	内容$
他$	配布$	サービス$	メニュー$
情報$	目次$	もくじ$	予定$
管理人$	一覧$	方法$	窓口$
案内$	名称$	写真$	種別$
ページ$	チャット$	コーナー$	ＣＨＡＴ$
ＢＢＳ$	著作権$	インフォメーション$	について$
戻る$	趣旨$	予約$	動画$
名$	から$	掲示板$	．$
,$	？$	！$	
.+と.+	.+・.+	.+, .+	.+ ／.+
.+ &.+			
.*ダウンロード.*	.*ログイン.*	.*更新.*	.*(.*

図 7　不適切だと思われる下位語候補のパターン

下位語候補集合獲得処理の前処理

HTML 文書中の表データも下位語候補集合を獲得するうえで重要なデータである．3.2 節で述べた下位語候補集合獲得処理を HTML 文書中の表データに適用すると，表データの行方向に関して下位語候補集合を獲得することになる．しかし，吉田ら (Yoshida, Torisawa, and Tsujii 2001) によれば表データ中に現れる属性（例えば「血液型」）に対するその値（A 型，B 型，AB 型，O 型）は，行方向ではなく列方向に並びやすいという結果が得られている．これは，表データ中の類似した要素は行方向ではなく列方向に並びやすいということを示している．そこで本研究では，下位語候補集合獲得処理により表データから意味的に類似した下位語候補集合を得るために，前処理として HTML 文書中に現れる表データの転置を行っている．これにより，HTML 文書中に現れる表データの列方向に関して下位語候補集合を得ることが可能になり，意味的に類似したより多くの下位語候補集合を表データから獲得することが期待できる．

下位語候補集合獲得処理の後処理

下位語候補集合獲得処理により獲得した下位語候補集合の要素間の意味的類似性をあげ，より共通の上位語を持ちやすい下位語候補，もしくはその集合だけを抽出するため，本研究では，下位語候補集合獲得処理の後処理として，獲得された下位語候補集合のうち，以下の条件に当てはまる下位語候補，もしくは下位語候補集合を削除する．

条件 1　文字列長が長い，もしくは文字種が頻繁に入れ替わる下位語候補

彼ら	物	あなた	ご覧	な	無料	必要
ほか	ぼく	僕	以下	一般	名	品
一部	下	下記	何	画面	夜	南
会	各種	株式	巻	関係	訳	日
基本	期間	気	系	個人	論	杯
向け	国際	最終	最新	妻	版	長
作	姿	子	私	誌	母	晩
事	事項	時間	次	自分	北	彼女
室	手	種類	集	所	味	東
書	女	女性	方法	詳細	無断	等
上	情報	状況	心	新	父	堂
人	人間	人気	西	先	武	内容
線	送料	多く	対象	沢	部	話
男	中心	昼	著	丁目	別	彼
目	誰	大人	子供	前半	編	番号
後半	だ	朝	登	ヶ	ゴール	クリック
メール	ＭＡＩＬ	ＵＲＩ	ＴＨＥ			

図 8　上位語として適切でない語

条件 2　図 7に示した正規表現パターンに適合する下位語候補

条件 3　要素数が 3 個以下，もしくは 20 個以上の下位語候補集合

条件 1 に当てはまる下位語候補を削除する理由は，下位語候補集合獲得処理において，下位語候補として獲得されてしまった「文」を削除するためである．下位語候補集合獲得処理は，単に HTML 文書中の表現が持つパスしか考慮していないため，同じパスを持つ「語」の他にも，同じパスを持つ「文」も獲得してしまう．しかし，ステップ 1 では下位語を獲得することを目的としているため，この処理で誤って下位語として獲得されてしまった「文」は削除する必要がある．そこで本研究では，文字列長が 12 以上の表現，もしくは文字種が 5 回以上入れ替わる表現を文として判断し，削除する．続いて，条件 2 に当てはまる下位語候補を下位語候補集合から削除する理由は，図 7に示した正規表現パターンに適合する下位語候補は，他の下位語候補と共通な特性を持ちにくいためである．図 7に示したパターンに適合する表現を削除することで，獲得された下位語候補間の意味的な類似性の向上が期待できる．最後に要素数が 3 個以下，もしくは 20 個以上の下位語候補集合を削除する理由は，要素数が 3 個以下の下位語候補集合については，各下位語候補間に意味的な類似性が見られにくいためであり，要素数が 20 を越える下位語候補集合に関しては，以降のステップにおいて処理に多大な時間がかかってしまうためである．

付録 B 上位語獲得に用いる不要語リスト

図 8に上位語として適切でない語の一覧を示す．図 8に示した不要語リストは，予備実験より得られた明らかに上位語にはなりにくい名詞，もしくは上位語として獲得されても価値の薄

いと考えられる名詞からなる.