HTML 文書からの属性語の自動抽出

徳永 耕亮　風間 淳一　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
{kosuke-t, kazama, torisawa}@jaist.ac.jp

1　はじめに

本論文では，Web 上の文書中から自動的に属性語を獲得する手法を提案する．本研究では，属性語とは，ある対象の特徴や構成要素など，その対象について知りたいユーザがまず要求するであろう項目を表す語であると仮定する．これらの語は，要約や情報抽出などの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．属性語は,「(対象語) の (属性語) は { 何，誰，どれ，いつ }?」のような質問が可能かによって判定できる．例えば，“北陸先端大” を対象語とすると，属性語は “所在地” “学長” などとなる．本研究では，広範な対象語の多様な属性語を統計量・構文パタン・係り受け・HTML タグを手がかりとして獲得する．これらの手がかりは対象語の上位語（上位概念）を検索語として Web から収集した文書から計算する．例えば，対象語 “北陸先端大” の上位語は “大学” である．このような上位語を用いることにより，対象語が属する意味クラスに共通する重要な属性が獲得できるようになり，また，データスパースネスを防ぐことができる．しかし，Web を対象とした場合，対象語の多くは通常のシソーラスには含まれないため，正しい上位語が得られるとは限らない．そこで，提案手法では新里ら [2] などの自動獲得手法により Web から獲得された上位語を利用する．

実験では，提案手法によって，34 個の単語クラスに対して 79.9%の精度で正しい属性語を獲得できることを示す．また，各手がかりの有効性や，上位語を用いることの有効性も検証する．

2　提案手法

本研究では以下に示す 3 つの仮説を立て，属性語の獲得に用いる．

仮説 1 属性語は対象語の上位語を含む文書に現れやすく，それ以外の文書には現れにくい．

仮説 2 属性語は HTML 文書中で強調表示されたりリストや表の要素になり易い．

仮説 3 属性語は対象語の上位語との間に，助詞 “の” を介した固有のパタン・係り受け関係を持つ．

本手法の属性語獲得は，以上の仮説に基づいて以下の 2 ステップで行われる．

ステップ 1 対象語の属性語候補集合の獲得

ステップ 2 構文パタン・HTML タグ情報・係り受けの頻度・統計量に基づいたスコアによる属性候補の順位づけ

仮説 1・2 は属性語獲得におけるステップ 1，仮説 1・2・3 はステップ 2 中でそれぞれ利用される．以下，ステップ 1・ステップ 2 の詳細を述べる．

2.1　属性語候補の獲得

ステップ 1 では，対象語の上位語や HTML タグの情報を利用して属性語の候補になる集合を獲得する．まず，対象語を下位語に持つ上位語を新里らが提案した獲得手法 [2] を用いて Web から獲得しておく．次に，前述の仮説 1・2 に基づき以下の 2 種類の文字列の集合を属性語の候補として獲得する．

(1) 上位語を含む Web 文書から $df \cdot idf$ によって獲得された単語集合．

(2) 上位語を含む Web 文書中でタグに囲まれた文字列の集合．

(1) の単語集合は，新里らの上位語獲得手法 [2] に類似した方法で獲得を行う．その理由は仮説 1 が新里らの手法の仮説の 1 つと性格が類似しているためである．具体的には，対象語の上位語 h を検索語として獲得した文書集合 $LD(h)$ 中の名詞[1] a を

$$C(h,a) = df(a, LD(h)) \cdot idf(a, G)$$
$$idf(a, G) = \log \frac{|G|}{df(a, G)} \qquad (1)$$

で定義されるスコアにより順位づけする．ここで，G は Web から無作為に収集しておいた大量の文書集合である．そして，スコアの上位 K 個を出力の単語集合とする．実験では $K = 100$ として評価を行う．

(2) は仮説 2 に基づいており，HTML 文書を改行・空白・数字・アルファベット・記号などを取り除いて連続した 1 行にしたときに，HTML タグで囲まれている N 文字以内の文字列を獲得する．実験では $N = 20$ とする．開始タグと終了タグは同じタグである必要はない．つまり，HTML タグをセパレータとして HTML 文書を分割したときに N 文字以下の要素を獲得する

[1] 普通名詞・サ変名詞・地名

<B>タイ風・カレー</B>
材料
鶏肉 400g，なす 2 個，バイマックルー 2 枚，ナンプラー大さじ 1.5
赤唐辛子 1.5 本，砂糖小さじ 1，ココナッツミルク，バジル<P>スパイス
コリアンダー，クミン<P>作り方
<OL><LI>材料をペースト状にして，カレーペーストを作る</LI><LI>カレーペーストにを熱した鍋に加えて香りを …

図 1 HTML 文書の例

表 1 属性語獲得の為の構文パタン

h の a は	h の a で	h の a が	h の a まで
h の a を	h の a から	h の a に	h の a より
h の a へ	h の a、		

ことに相当する．これにより，HTML 文書中で強調表示されたり，リストや表の要素になり易い文字列を抽出することができる．例えば，図 1 に示す HTML 文書からは，“タイ風カレー”，“材料”，“スパイス”，“コリアンダー クミン”，“作り方” が獲得される．

以上の方法で獲得された (1)，(2) の集合の和を属性語候補集合，その要素を属性語候補と呼ぶ．

2.2 属性語候補の順位づけ

ステップ 2 では，ステップ 1 で獲得した属性語候補を仮説 1・2・3 を反映するスコアにより順位づけし，上位の M 個を属性語として獲得する (実験では $M = 20$)．上位語が h であるような属性語候補 a のスコアは以下の式で計算される．

$$V(h,a) = n(h,a) \cdot t(h,a) \cdot f(h,a) \cdot idf(a,G) \qquad (2)$$

まず，式 2 中の $n(h,a)$ は，仮説 3 を反映したスコアで，前述の上位語 h を検索語としてダウンロードした文書集合 $LD(h)$ 中で，上位語 h と属性語候補 a が表 1 に挙げるパタンのいずれかで共起した回数である．上位語と属性語がこれらのパタンで共起するのは自然であり，a が正しい属性語の場合 $n(h,a)$ が大きくなる事が期待できる．例えば，図 1 の HTML 文書の例から獲得される属性語候補の中で，“材料” は「カレーの材料」と言われることが多いので $n(h,a)$ が大きくなり，逆に，“タイ風カレー” は「カレーのタイ風カレーは」などと言われることは少ないので $n(h,a)$ が小さくなる．

次に，$t(h,a)$ は，仮説 2 を反映したスコアで，上位語 h を検索語としてダウンロードした文書集合 $LD(h)$ 中で，属性語候補 a が HTML タグで囲まれている頻度である (定義は属性語候補の獲得の際と同様)．このスコアは，属性語候補の獲得において (1) で獲得された候補に対しても HTML タグの情報を与えて正しい候補を選択するのに役立つのはもとより，(2) の HTML タグに基づいて獲得された候補から正しい候補を選択するのにも役立つ．タグを用いた属性語候補の獲得には，幅広い属性語候補の抽出が可能であるという利点がある一方，非属性語も含まれやすいという欠点もある．しかし，そのような非属性語は多くの場合，文書集合 $LD(h)$ において繰り返し出現する事は少なく，対象語に特化した文字列である．つまり，HTML タグによって囲まれる頻度を比べた場合，属性語は高い値を獲得し，非属性語は低い値を獲得することになると考えられ，結果として属性語の獲得精度向上に貢献すると考えられる．

また，$f(h,a)$ は，構文解析された新聞記事 33 年分[2]中で上位語 h が助詞 “の” を介して属性語候補 a に係る頻度である．このスコアは，$n(h,a)$ 同様仮説 3 を反映したスコアであるが，表 1 のパタンとは異なり属性語候補の後にくる助詞は何でも良い点，マッチングを構文解析された新聞記事に対して行っている点が異なる[3]．ここで，新聞記事から得る $f(h,a)$ も用いる理由は，マッチング対象の文書量を増やし，より信頼性の高いスコアを獲得するためである．これは，Web が最大のコーパスであることを考えると矛盾しているように思われるかもしれない．しかし，通常の商用検索エンジンでは，検索語を含む文書のごく一部（上位 1000 文書程度）しか実際には得られない．そのため，現実問題として Web から上位語を含む大量の文書を収集することはできず，新聞記事での頻度の方が信頼性が高いということが起こる．なお，検索エンジンに構文パタンを検索語として得られるヒット件数を利用することも考えられるが，本研究でそうしないのは，実験で述べるように，属性語の候補が各クラスに平均で約 2 万個も存在し，検索要求回数が多くなって検索エンジンへの負担が大きくなり過ぎるからである．

最後に，$idf(a,G)$ は，属性語候補集合を求めた際に用いた式 (1) で定義される値に，値がゼロにならないように 1 を加えたものである．これは，仮説 1 に対応する．

なお，スコアの各項のうち $n(h,a)$, $t(h,a)$, $f(h,a)$ は頻度がゼロの場合には 10^{-3} という小さい値を与える．また，$idf(a,G)$ は定義から最小値は 1 になる．

以上の 2 ステップを経て，本研究で最終的に出力する属性語が獲得される．

3 実験・評価

3.1 提案手法による属性語獲得の評価

この実験では，34 個の単語クラスに対して属性語を抽出して評価を行う．この 34 個の単語クラスは以下のようにして選択した．まず，新里らの方法 [2] で獲得された 1,589 個の上位語（で定義される単語クラス）から開発セット用のクラスを除く．その上で，

- 下位語が 10 個以上存在し，全ての下位語に対し

[2]読売新聞 1987-2001，毎日新聞 1991-1999，日経新聞 1990-1998; 計 3.01GB

[3]係り受け解析された大量の新聞記事から候補の後にくる助詞を無視して掛り受け頻度を求めた統計データが利用可能であったためである．

表 2 提案手法で獲得された属性語の例

上位語	対象語（下位語）の例	属性語（上位 **20** 個）
カレー	グリーンカレー レッドカレー マサマンカレー パナンカレー イエローカレー	作り方*, 材料*, レシピ* メニュー, 値段*, 歴史* 仕上げ*, 名前*, 辛さ* 料理, 種類*, ホームページ* レトルト, 香り*, お店* 具*, オリエンタル, 店* ソース, スパイス*
病院	横浜労災病院 横浜船員保険病院 横浜市立市民病院 有馬病院 清水丘病院	ホームページ*, 施設*, 理念* 情報*, 紹介*, 認定*, 精神科 医師*, 電話*, 名前*, 検査* 対応*, 薬局*, 所在地*, 治療* 職員*, 院長*, 機能*, 住所* 診察*

“*” は提案手法により獲得された妥当な属性語を示す.

て妥当である 20 個の上位語（この条件を満たす上位語は 20 個しかない.）

- 下位語が 4 個以上存在し，誤った下位語が 1 つ以下であり，[2] でのスコア上位の 14 個の上位語

による 34 個の単語クラスを選択した.

なお，文書のダウンロードには検索エンジン goo[4]を用いた．上位語を検索語として獲得された文書 $LD(h)$ の文書数はクラス平均で 857 個であった．また，文書集合 G は [2] 中の 10^6 件の Web 文書と同一のものを用いた．ステップ 1 で獲得された属性語候補の数は，クラス平均で約 2 万個であった.

獲得された上位 20 個の属性語を， 属性語の定義から導かれる以下の基準に従って著者の一人が正誤を判定した.

判定基準 {この，その} (上位語) の (属性語) は {何，誰，どれ，いつ}？のような質問に対して答えが（仮想的にでも）想像できる.

また， 属性語の語尾に以下の語を補うことで上の基準を満たす場合も正しいと判定した.

者・名・数・時間・方法・時期

これは,「この病院の診察は午前 9 時から午後 5 時です」の例にみられるように，属性語の基準を満たす表現（この例では “診察時間”）が語尾を省略して使用されることが多く，誤りとするのは適当でないと考えたからである．また，これにより判定の揺れを大きく押さえることができる．それでも判定が難しい場合には，判定基準の上位語をクラス中の具体的な下位語に換えて答えが想像できる場合も正解と判定した．獲得された上位 20 個の属性語の内の正しい属性語の割合 (適合率) を用いて評価を行う.

実験の結果，34 クラスの平均の適合率は 79.9% (543/680) となり，十分に高い精度で属性語が獲得できることが確認できた．表 2 に，実際に抽出された属性語の例を示す.

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

提案手法では，属性語を Web から獲得するため，非常に多くのクラスに共通して

名前・名称・詳細・情報・概要・紹介・歴史・特長
データ・特色・ホームページ・サイト・特徴・ポイント

といった属性語が獲得されることが分かった． これらは正しい属性語ではあるが，あらかじめ列挙しておくことも可能であるので，これらの属性語以外の自明でない属性語の獲得精度も求めた．その結果，平均の適合率は 75.4%（428/568）となり，自明でない属性語も十分獲得できていることが確認された.

3.2 順位づけの各スコアの有効性

この実験では， 各仮説あるいは対応する式 (2) 中の各スコアの妥当性を検証するため， 式 (2) から各スコアを除いた式で順位付けした場合の精度を求めた．ただし仮説 2（HTML タグ）の有効性を調べる際には，属性語候補を求める時点でもタグは用いず， スコアからも $t(h,a)$ を抜いて実験した.

まず，$n(h,a)$ を抜いた場合には，適合率の平均は 3.9%低下した．同様に，$t(h,a)$ を抜いた場合には，適合率の平均は 14.3%低下した．また，$f(h,a)$ を抜いた場合には，適合率の平均は 14.1%低下した．最後に，$idf(h,a)$ を抜いた場合には，適合率の平均は 1%低下した.

これらの結果から，各仮説（各スコア）は精度の向上に有効に働いていることがわかる．特に，タグによる頻度と，新聞記事から獲得した係り受け頻度が有効であった．タグによる頻度を抜いた場合は，属性語候補が統計量を用いる方法のみで獲得されるため，正しい属性語が属性語候補中に含まれる割合が低下した事が大きな理由と考えられる．局所文書集合中のパタン頻度 n が， 新聞記事における係り受け頻度 f と比べて効果が小さかったのは，マッチング対象の文書量が少なかったためと考えられる．これは，今後検索制限のない検索エンジンを使用できるようになれば，獲得精度がさらに向上することを示唆している．idf を抜いた場合に，ほとんど変化が見られないのは，属性語候補となるような HTML タグに囲まれる語や統計量 $(df \cdot idf)$ が大きい語は $idf(h,a)$ が大きく， 既に同様の効果が織り込まれているためと考えられる.

3.3 上位語の有効性

最後に，提案手法で上位語を用いる優位性を検証する．上位語を用いない場合の一つの方法として，上位語で定義されるクラス中の語（下位語）を利用することが考えられる．考え得る実験の設定は多岐にわたるが，ここでは，属性語候補を獲得する文書集合を上位語ではなく下位語を検索語として収集し，順位づけの際に上位語の構文パタンの頻度 $n(h,a)$ の代わりに $\sum_i n(l_i,a)$（h を各下位語 l_i で置き換えて和を取った

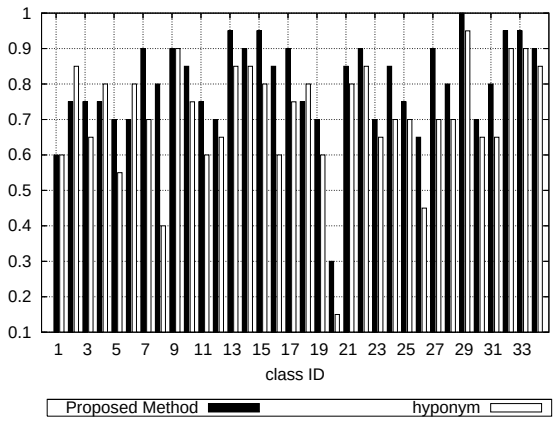

図 2 上位語を用いた場合 (提案手法) と下位語を用いた場合の各クラスの適合率.

もの）を用いる手法を比較手法として，上位語を用いる提案手法との精度の差を検証した．なお，下位語で収集できた文書数が，上位語で収集できた文書数を上回る場合には，ランダムに削除して文書数を合わせた．

実験の結果，提案手法と比較して適合率の平均は 9.0%低下することが確認された．図 2 に，このときの各クラスに対する適合率を示す．

次に，文書集合の収集は下位語のままで，構文パタン頻度を $n(h,a) + \sum_i n(l_i,a)$ として上位語も利用するようにした場合の精度を調べた．この場合には，適合率の平均の低下は 5.1%と軽減された．これらの結果から，文書集合を求めるクエリに上位語を用いることの優位性と，構文パタンに上位語を用いる有効性が確認できた．今後さらに，文書の収集にも上位語と下位語を両方使う場合などの検証も行っていきたい．

4 関連研究

これまでにも，文書から属性や (属性, 属性値) の組を獲得する研究はいくつか行われている．吉田ら [4] の Web 上の表の解析・クラスタリング・統合の手法は，その過程で，属性語を獲得する．しかし，属性語獲得自体の性能評価は行われていない．また，獲得対象が表に限られているため，提案手法のように広範なクラスの属性語が獲得できるとは限らない．また，吉田ら [3] は，[4] で獲得された属性語のオントロジを基に，箇条書きなど表に限定されない表現中の属性・属性値の出現を判別する HMM を学習する手法を提案している．しかし，この HMM は [4] の手法で表から獲得された属性・属性値しか判別できないため，本研究の目的とは合致しない．加えて，[4][3] では，統計情報や構文パタンなどを用いていないという点で本研究とは異なる．高橋ら [5] は質問応答のための (対象物，属性名，属性値) という三つ組みの属性関係をテキスト中から獲得する手法を提案している．統計情報や構文パタンを用いる点では提案手法と共通するが，対象物を固有名詞に限定している点，属性の候補をあらかじめ人手で選択している点が本研究と大きく異なる．また，Fleischman ら [1] の手法では，まず構文パタンを用いて属性・属性値の抽出した後，教師あり学習で構築されたモデルによってフィルタリングを行い高精度での獲得を実現している．しかし，使用されている構文パタンは人名に関するものに限定されているため，本研究のように多くの属性語を種類に限らず自動で抽出するのは難しいと考えられる．紙面の都合で詳しくは述べないが，本研究では，提案手法中の各スコアを単純に素性として学習した SVM によるフィルタリングを試みたが，精度の向上は見られなかった．その点で，Fleischman らのフィルタリングモデルで用いられた様々な種類の素性は今後の精度向上のために参考になると考えられる．

5 まとめと今後の課題

本稿では，Web 文書から，対象語の一般的に重要な属性語を獲得する方法を提案し，実験により有効性を示した．実験では，さらに，用いられた各々の手がかりの有効性や，文書集合を求める検索語として上位語を用いることの優位性も確認した．

今後の課題としては，まず，複数人による評価実験がある．今回の実験では著者の一人が評価を行ったが，人による評価のゆれも考慮した上で，提案手法の有効性を確認する必要がある．また，今回は属性語の獲得を行ったが，獲得された属性語の値を各対象について Web から抽出する手法を開発し，最終的にはユーザが入力した対象語についての要約を出力するシステムを構築したいと考えている．

参考文献

[1] M. Fleischman, E. Hovy, and A. Echihabi. Offline strategies for online question answering: Answering questions before they are asked. In *Proc. of ACL03*, pages 1–7, 2003.

[2] K. Shinzato and K. Torisawa. Acquiring hyponymy relations from web documents. In *Proc. of HLT-NAACL04*, pages 73–80, 2004.

[3] M. Yoshida, K. Torisawa, and J. Tsujii. *Web Document Analysis*, chapter 10 (Extracting Attributes and Their Values from Web Pages). World Scientific, 2003.

[4] M. Yoshida, K. Torisawa, and J. Tsujii. Integrating tables on the world wide web. 人工知能学会論文誌, 19(6):548–560, 2004.

[5] 高橋 哲朗　乾 健太郎　松本 裕治. テキストから属性関係を抽出する. 情報処理学会研究報告 *(2004-NL-164)*, pages 19–24, 2004.