Webからの単語クラスの簡単な作り方

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

1　はじめに

従来より単語クラスの自動獲得に関する研究は数多く行われてきた [1, 2, 3]．しかしそのほとんどは名詞と名詞，または名詞と動詞といった単語間の共起関係を利用するものであり，単語クラスの獲得にはそれなりの時間を要する．一方で我々は，HTML 文書中に現れる箇条書きや表などのレイアウトには，要素同士が意味的に類似しているものがあることを示した [4]．そこで本稿では，この HTML 文書より得られるレイアウト情報と，従来より用いられてきた単語の共起関係を組み合わせることで，簡単かつスピーディに単語クラスを獲得する手法について述べる．本稿では，“要素同士が意味的に類似している単語または複合語の集合” を単語クラスと呼ぶ．もし仮に HTML 文書中に現れる箇条書きや表の要素同士が意味的に類似しているかどうかを直ちに判別できれば，今も尚急速に増え続ける HTML 文書より，大量の単語クラスを獲得することが可能になる．獲得された単語クラスの応用としては，例えば我々が以前に提案した上位下位関係獲得手法 [4] への組み込みが考えられる．この手法は，箇条書きや表に含まれる表現に対し，それら各表現に共通する上位語を求め上位下位関係の獲得を行う．そのため，提案手法により単語クラスと見なされた箇条書きや表だけを入力として与えれば，精度の向上が期待できる．

本研究では図 1 に示した仮説 1 に従い，HTML 文書中に現れる箇条書きや表といった同一のレイアウトに現れる表現の集合を，関連する表現の集合として獲得する．続いて仮説 2 に基づき，抽出された集合の各要素間の共起の強さを計算し，集合中の表現同士が意味的に類似しているかどうかの判定を行う．その後，類似していると判断された表現の集合を単語クラスとして獲得する．具体的には，要素間の共起の強さとして検索エンジンより得られるヒット件数を使って求めた相互情報量を用いる．そして，相互情報量を素性として Support Vector Machine(SVM)[5] に与え，単語クラスかどうかの判別を行う．この時，素早く単語クラスの判定を行うための工夫として，n 個の表現から成る集合に対しては，ランダムに選択した n 組の表現組に限り相互情報量を求める．

実験の結果，HTML 文書から単語クラスを獲得する手法の一種とみなせる [4] に比べ，高い精度で単語クラスの獲得を行えることがわかった．

2　関連研究

HTML 文書から単語クラスを獲得する研究としては，我々が以前提案した上位下位関係獲得手法（以降，Hyponymy Relation Acquisition Method を略して HRAM と呼ぶ）[4] がある．HRAM は HTML 文書中に現れる箇条書きや表を単語クラスと見なし，それらに含まれる表現に共通する上位語を求めることで上位下位関係を獲得する．しかし，全ての箇条書きや表が意味的に妥当な単語クラスと見なせるわけではない．そこで，HRAM では獲得された上位語と単語クラスの要素との係り受け関係及びヒューリスティックルールを用い，妥当な上位語が獲得されやすい単語クラスを優先的に出力するというフィルタリングを行っている．フィルタリングには，検索エンジンを利用してクラス中の要素を含んでいる HTML 文書を 1 要素当り 100 件収集し，そこから各要素の持つ係り受け関係を求めるという比較的重い処理を要する．それに対し提案手法は，検索エンジンが返すヒット件数だけを基に単語クラスの意味的一貫性を判定するため，HRAM に比べ素早く単語クラスの意味的一貫性を測ることができる．

仮説 1 HTML 文書中で同一のレイアウトに現れている表現は互いに関連している

仮説 2 意味的に類似した表現同士は共起しやすい

図 1 本研究で用いている仮説

一方，新聞記事などの HTML 文書以外の文書を対象とした単語クラスの自動獲得に関する研究は数多くある [1, 2, 3]．ここでは代表的なものについて触れる．Church ら [1] は単語の出現頻度を基に 2 単語間の相互情報量を求めることで，$\langle doctors, nurses \rangle$ のような意味的に類似した単語の組を獲得している．しかし，相互情報量だけでは単語同士が関連を持つということしかわからないため，$\langle doctor, bills \rangle$ のような関連はあるが意味的に類似していない組も獲得されてしまう．本研究でも相互情報量を単語クラスの要素間の意味的類似度として用いているが，“同一レイアウトの要素である” という制約を用いることで $\langle doctor, bills \rangle$ のような組の排除を狙う．

その他では，単語の係り受け関係を用いて単語クラスを獲得する研究がある．Lin[2] は係り受け関係から単語間の意味的類似度を求め，類似した単語同士をまとめることで単語クラスを獲得している．また Rooth[3] は，係り受け関係及び EM 法を用い，単語がクラスへ属する確率を推定し，単語クラスの獲得を行っている．本研究は，大量の構文解析済コーパスを必要としないという点でどちらの手法とも異なる．

3　提案手法

本研究では，先述したように図 1 に示した仮説に従い，後述する 2 つのステップを経ることで単語クラスの獲得を行う．

3.1　関連表現集合の獲得（ステップ 1）

ステップ 1 では，仮説 1 に従い HTML 文書中に現れる各表現の持つ**パス**に注目することで，意味的に関連した表現の集合を獲得する．ここでパスとは，HTML 文書中の表現を

囲んでいるタグをそのネストの順序に従って，リスト形式で表したものである．例えば，図 2 (A) に示した HTML 文書中の各表現は，

<LI>取り扱い店舗一覧</LI>
<UL><LI>Tower Records</LI> <LI>HMV</LI>
<LI>レコファン</LI> <LI>新星堂</LI></UL>

のようにタグ付けされているため，それぞれ以下のようなパスを持っていると考えられる．

{(LI)，取り扱い店舗一覧 },
{(UL，LI)，Tower Records},
{(UL，LI)，HMV},
{(UL，LI)，レコファン },
{(UL，LI)，新星堂 }

ステップ 1 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ることで，それらを互いに関連のある表現の集合として獲得する．例えば，先程の HTML 文書からは，

A　{Tower Records, HMV, レコファン, 新星堂 }

といった表現の集合が獲得される．本研究では，この集合のことを**関連表現集合**と呼ぶ．関連表現集合の獲得は，特定の HTML タグに注目して行っているわけではないため，箇条書き以外のレイアウトからも獲得が可能である．例えば，図 2 (B) の HTML 文書からは以下のような関連表現集合が獲得される．

B　{ 当日, 1 週間, CD, 190, 290, ビデオ, 240, 340}

ステップ 1 で獲得される関連表現集合には，B のような要素同士に意味的類似性が見られず単語クラスとしては見なすことができない集合が含まれていることに注意されたい．

3.2 SVM を用いた関連表現集合の意味的類似性の判定 (ステップ 2)

ステップ 2 では，ステップ 1 で獲得された関連表現集合から “要素同士が意味的に類似している” と判断される集合を単語クラスとして獲得する．そのため，仮説 2 に基づき関連表現集合中の表現同士の共起の強さを求め，それを基に関連表現集合が単語クラスかどうかの判定を行う．より具体的には，表現同士の共起の強さ表す指標として，共起頻度と相互情報量を用いる．そして，それらを素性として SVM[5] に与え，関連表現集合を単語クラスとそうでないものとに分類する分類器を生成する．

共起頻度及び相互情報量を求めるため，まず n 個の表現から成る関連表現集合中から，表現の組を n 組生成する．具体的には，関連表現集合中の表現 e について，e とは異なる表現 e' をランダムに選び組 $\langle e, e' \rangle$ を生成する．この操作を関連表現集合中の全ての要素について行う．例えば，先程示した関連表現集合 A の場合，以下のような表現の組の集合が生成される．

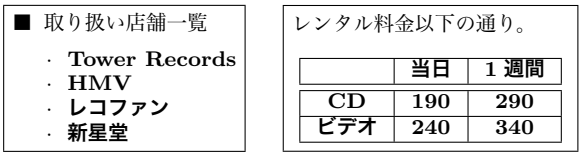
{⟨Tower Records, 新星堂 ⟩, ⟨HMV, レコファン ⟩,
⟨ レコファン, Tower Records⟩, ⟨ 新星堂, HMV⟩}

そして，生成された組について共起頻度及び相互情報量を求める．表現 e と e' の共起頻度 $docs(e, e')$ 及び相互情報量 $I(e, e')$ は以下の形で与えられる．

$$docs(e, e') : \text{表現 } e \text{ と } e' \text{の両方を含む文書数}$$

$$I(e, e') = \log_2 \frac{\frac{docs(e, e')}{N}}{\frac{docs(e)}{N} \times \frac{docs(e')}{N}}$$

ここで $docs(e)$ は表現 e を含む文書数である．本研究では，$docs(e)$，$docs(e, e')$ として，検索エンジン goo[1] より得られる検索ヒット数を用いる．また N は全文書数を表しており，本研究では goo が検索対象としている URL の総数である 4.20×10^{10} としている[2]．共起頻度及び相互情報量を全ての組 (つまり，$n \times (n-1)/2$ 組) ではなくランダムに選択した n 組に限るのは，単語クラスの判別を素早く行うための工夫である．全ての組について計算しないことにより，単語クラス獲得精度の低下が懸念されるが，実際にはそれほど低下しないことを実験により示す．



(A) 箇条書きレイアウト　　(B) テーブルレイアウト

図 2 HTML 文書の例

本研究で SVM に与えた素性の一覧を表 1 に示す．共起頻度，相互情報量の他にも，関連表現集合の要素数，集合に含まれる表現単独のヒット件数 $docs(e)$ なども素性として用いている．これは，与えられた関連表現集合が単語クラスかどうかを判別する際，これらも重要な手がかりになると考えたためである．また，全ての表現や表現の組について，その文書頻度，共起頻度，相互情報量を素性として用いた方が良いと思われるかもしれない．しかし，ステップ 1 で獲得される関連表現集合の要素数 n は $4 \leq n \leq 30$ と一定でないため，全て表現や表現の組についてその文書頻度 $docs(e)$ や共起頻度 $docs(e, e')$，相互情報量 $I(e, e')$ を同時に素性として利用することは難しい．そのため本研究では，それぞれの値の最大値，2 番目に大きな値，最小値，2 番目に小さい値の 4 種類だけを素性として SVM に与えている．

また提案手法では，SVM により得られる超平面との距離を “単語クラスらしさ” として捉え，単語クラスと見なされた関連表現集合を，この距離に従い降順にソートし出力する．

4 実験

実験は，HTML 文書から単語クラスを獲得する手法の一種である HRAM との比較及び，全ての組について共起頻度と相互情報量を求めた場合との比較の 2 種類行った．以下，実験の設定及び両実験結果について述べる．

4.1 準備

実験に伴い，1.00×10^6 件の HTML 文書 (10.5 GB，タグ付き) を Web より収集し，それらに対し 3.1 節で述べたステップ 1 を適用した．その結果，161,597 個の関連表現集合を得た．そして，その中からランダムに 2,000 個を選び，うち 500 個を学習データ，1,500 個を評価データとした．続いて集合の各要素を JUMAN[3] により形態素解析し，助詞を含む要素を持つ関連表現集合を各データセットから削除した．この理由は，助詞を含んでいる要素は不適切な要素である可能性が高いと考えたためである．その結果，学習データは 343 個に，評価データは 1,001 個にそれぞれ減少した．

本研究では，フリーウェアである *TinySVM*[4] を使用し分

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

[2] http://help.goo.ne.jp/door/

[3] http://www.kc.t.u-tokyo.ac.jp/nl-resource/juman.html

[4] http://chasen.org/~taku/software/TinySVM/

表 1 本研究で用いている素性

番号	概要
1	C の要素数
2	文書頻度 $docs(e)$ が 0 の要素数
3	文書頻度 $docs(e)$ の平均
4	文書頻度 $docs(e)$ の総和
5	C 中で最も大きい文書頻度 $docs(e)$
6	C 中で 2 番目に大きな文書頻度 $docs(e)$
7	C 中で最も小さい文書頻度 $docs(e)$
8	C 中で 2 番目に小さい文書頻度 $docs(e)$
9	共起頻度 $docs(e, e')$ が 0 になる組数
10	共起頻度 $docs(e, e')$ の総和
11	共起頻度 $docs(e, e')$ の平均
12	相互情報量 $I(e, e')$ の総和
13	相互情報量 $I(e, e')$ の平均
14	P 中で最も大きい共起頻度 $docs(e, e')$
15	P 中で最も大きい相互情報量 $I(e, e')$
16	P 中で 2 番目に大きい共起頻度 $docs(e, e')$
17	P 中で 2 番目に大きい相互情報量 $I(e, e')$
18	P 中で最も小さい共起頻度 $docs(e, e')$
19	P 中で最も小さい相互情報量 $I(e, e')$
20	P 中で 2 番目に小さい共起頻度 $docs(e, e')$
21	P 中で 2 番目に小さい相互情報量 $I(e, e')$

C: 関連表現集合，P: C から生成された表現の組の集合

類器を生成した．カーネル関数は，学習データを用いた 4 分割交差検定法で最も精度の高かった 2 次の ANOVA カーネルを用いている．また，学習データは後述するラベル付け基準に従って筆者が単独でラベル付けを行った．

次に評価基準について述べる．単語クラスを明確に定義することは難しい問題であるが，本研究では仮に以下の基準を満たすものを単語クラスとして見なした．

ラベル付け基準 関連表現集合中の 7 割以上の要素に共通する具体的な上位語を考えることができれば，その集合を単語クラスとしてみなす．ただし，考えられる上位語として “物” や “事” などの一般的過ぎる語は除く．

“一般的すぎる語” の判断は評価者の主観に委ねられるため，この評価基準では不十分に感じられるかもしれない．そのため，間接的ではあるが HRAM を用いて，獲得された単語クラスを評価する評価基準を新たに設けた．以下，この評価基準を**間接評価基準**と呼ぶ．この基準は，“精度良く単語クラスが獲得されているならば，クラスを用いている他の NLP アプリケーション（ここでは HRAM）の性能が改善されるだろう” という考えに基づいている．間接評価基準では，提案手法により獲得された単語クラスを用いることで，上位下位関係の獲得精度がどの程度向上するかを見ることにより，単語クラスを評価する．この評価基準において，単語クラスを用いた場合とそうでない場合との間に差が見られるようであれば，ラベル付け基準の妥当性をある程度示すことができると考えられる．

4.2 単語クラスの獲得実験

評価用データに対して，提案手法及び HRAM を適用し評価実験を行った．評価基準としてラベル付け基準に従った場合の実験結果を図 3 に示す．図中の Y 軸は出力された単語クラスの精度を，X 軸は単語クラスの数をそれぞれ示している．提案手法の出力は，SVM により単語クラスと判断された関連表現集合 325 個であった．これらは，超平面との距離

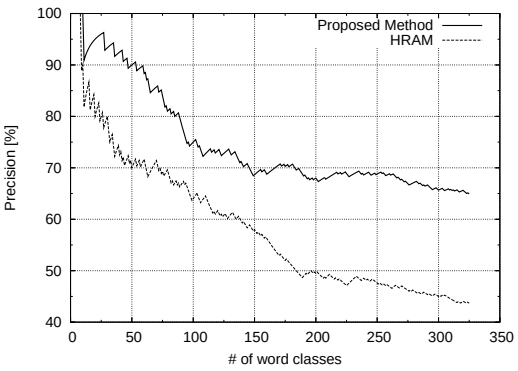


図 3 単語クラスの獲得精度の比較

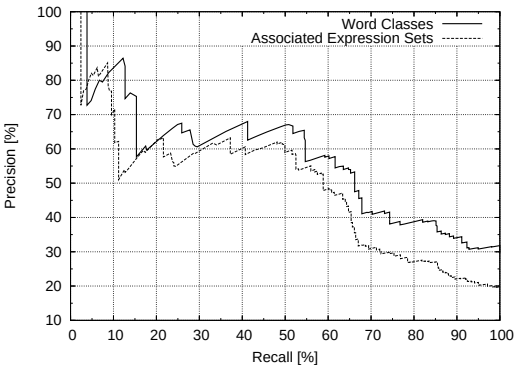


図 4 上位下位関係獲得の P-R 曲線

に従い降順にソートされている．その一方で HRAM の出力は，フィルタリングの結果残った 518 個の単語クラスであった．比較のため，HRAM により獲得された単語クラスから 325 個選びだした．具体的には，HRAM が出力するスコアに従い単語クラスをソートし，その上位 325 個を比較対象として選んだ．図より提案手法の方が HRAM に比べ高い精度で単語クラスを獲得できていることがわかる．その精度は，上位 100 クラスを出力とした場合で約 75 %，上位 200 クラスの場合では約 68 %を示している．

ここで，単語クラスの判別に要する時間について考察する．提案手法では，表 1 に示した素性を用いて単語クラスの判別を行っているが，これらの素性を生成するためには，$docs(e)$ 及び $docs(e, e')$ を求めるだけでよい．そのため，検索エンジンへの問合わせに要する時間を RT，クラスの要素数を n とすると，$2nRT$ の時間で単語クラスの判別を行うことができる．一方 HRAM では 1 要素ごとに，(1) 検索エンジンへの問合わせ，(2) 検索結果上位 100 文書のダウンロード，HTML 文書中に含まれる文の (3) 形態素解析，(4) 係り受け解析，の各処理を行う．そのため，1 文書当り平均で m 文含まれているとすると，$(RT + 100DT + 100mMT + 100m\alpha PT) \times n$ だけの時間を単語クラスの判別に要する．ここで，DT は 1HTML 文書をダウンロードする時間，MT は 1 文を形態素解析する時間，PT は 1 文を係り受け解析する時間である．また，α は文中に関連表現集合中の要素が現れる確率である．HRAM では各要素の持つ係り受け関係だけが必要なため，それらを含む文のみを対象に係り受け解析を行っている．仮に，m=30，RT=5 sec，DT=1 sec，MT=0.1 msec，PT=1 msec，α=0.05 とすると，HRAM は単語クラスの判別に $105.45n$ sec 要するのに対し，提案手法は $10n$ sec で判別できることになる．以上より，大雑把であるが提案手法は HRAM に比べ高速に単語クラスの判別を行えると言える．

CLASS 676: 食品会社
ブルボン, 駿河屋, ピエトロ, 和光堂, セイヒョー, はごろもフーズ, 理研ビタミン, 井村屋製菓, 雪印種苗
CLASS 614: 鳥取県気高町の特産品
どんどろけ飯, 三五八漬, 荒磯焼, 三善みそ, 塩さば, はま茶, 貝がら節和紙折り紙人形, 貝がらもなか
CLASS 1901: AFO 1998 出演アーティスト
香西かおり, 梅津和時, 賈鵬芳, 大工哲弘, 仙波清彦, 高橋淑子, 佐藤一憲, 中川博志, 矢野晴子, 深見邦代
CLASS 1331: ハンドルネーム
ユウノ, 時陰, 蒼幻, 涼杏, 異龍闇, 間宮蛍, 早麻キリア
CLASS 1535: 学校行事
運動会, 卒業式, 謝恩会, 町民運動会, 夏休み作品展, 陸上記録会, 臨海学校

図 5 獲得された単語クラスの例

次に間接評価基準に従った場合の実験結果を図 4 に示す．この図は，提案手法により獲得された 325 個の単語クラスを HRAM の入力として用いた場合とそうでない場合（つまり，1,001 個の関連表現集合を入力とした場合）の HRAM の P-R 曲線である．実際に獲得された妥当な上位下位関係数は異なっているため，各々の場合について最終的に獲得された妥当な上位下位関係数を基に Recall を計算している．つまり，図 4 は同数の単語クラスと関連表現集合を HRAM の入力として与えた時の精度の差を示している．この図より単語クラスを入力として与えた方が高い精度で上位下位関係を獲得できることがわかる．この結果から，ラベル付け基準の妥当性をある程度示せたと言える．

最後に提案手法が獲得した単語クラスの例を図 5 に示す．

4.3 ランダムに n 組を選択することの弊害

提案手法では，n 個の表現から成る関連表現集合に対して，3.2 節の手順で生成した n 組についてしか相互情報量を計算していない．しかし，全ての組 $(n \times (n-1)/2$ 組) について計算した方が，高い精度が得られそうである．そこで，全ての組について相互情報量を計算した場合と，n 組に限定した場合とで，どの程度精度に差が出るのかを確認するための比較実験を行った．この実験に限り，関連表現集合中の要素単独での文書頻度 $docs(e)$ 及び 2 単語の共起頻度 $docs(e, e')$ は，予め Web より収集しておいた 1.74×10^7 件の HTML 文書 (191 GB，タグつき) から求めた文書頻度で代用している．その理由は，全ての組について検索エンジンを利用してヒット件数を求めると，検索エンジンに対して多大な負荷をかけてしまうためである．比較実験の結果を図 6 に示す．評価基準には，ラベル付け基準を用いている．図中の Exhaustive Pairs は全通りについて相互情報量を計算した場合，Random Pairs は n 組に限定して相互情報量を計算した場合である．Random Pairs については，偶然に高い精度が得られているということが考えられるため，10 回行った実験の平均を示した．検索エンジンから HTML 文書集合に変えたことで，上位 1〜100 クラスを出力データと見なした時で，15〜5％精度が低下したが，100 クラス以降についてはだいたい 3％以内であった．Exhaustive Pairs と Random Pairs の差は上位 100 クラスを出力データと見なした時が最大で，約 10％の差が見られる．しかし，それ以外の部分では概ね 5％以内に収まっている．全通りの相互情報量を計算するために 34,445 回検索エンジンに問い合わせる必要があるのに対し，提案手法では 13,348 回の問い合わせですむことを考えれば，許容できる範囲ではないかと考えられる．

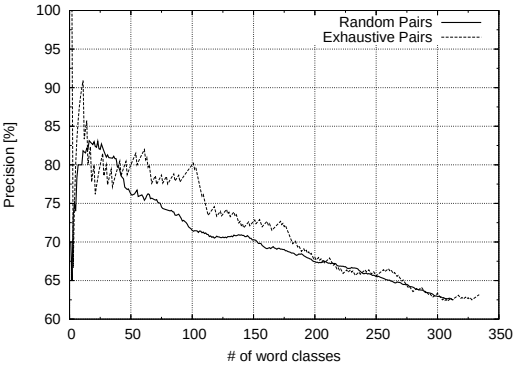


図 6 全ての組を考慮した場合との比較

5 おわりに

本稿では，HTML タグにより与えられるレイアウト情報及び相互情報量を用いて，HTML 文書から簡単かつスピーディに単語クラスを獲得する手法について述べた．より具体的には，HTML 文書中に現れる箇条書きや表といったレイアウトを単語クラスとして見なし，要素間の共起頻度及び相互情報量を用いて単語クラスとして適切かどうかの判定を行っている．実験の結果，HTML 文書より単語クラスの獲得を行う既存の手法に比べ，高い精度で単語クラスの獲得が行えることがわかった．今後は，より客観的な評価手法により獲得された単語クラスの評価を行いたいと考えている．

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

[1] Kenneth Ward Church and Patrick Hanks. Word association norms, mutual information, and lexicography. In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics*, pp. 76–83, 1989.

[2] Dekang Lin. 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, 1998.

[3] Mats Rooth, Stefan Riezler, Detlef Prescher, Glenn Carroll, and Franz Beil. Inducing a semantically annotated lexicon via em-based clustering. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pp. 104–111, 1999.

[4] Keiji Shinzato and Kentaro Torisawa. Acquiring hyponymy relations from web documents. In *Proceedings of Human Language Technology conference/North American chapter of the Association for Computational Linguistics annual meeting*, pp. 73–80, 2004.

[5] Vladimir Vapnik. *The Nature of Statistical Learning Theory*. Springer, 1995.