自 然 言 語 処 理 140－1
（2000. 11. 21）

日本語単語分割へのタグなしコーパスとタグ付きコーパスの利用

新納浩幸

茨城大学 工学部 システム工学科

自然言語処理の個々の問題を分類問題ととらえ，帰納学習の手法により問題解決を図る場合，最も重要な課題は，訓練データをどのように準備するかである．ここでは具体的な問題として日本語単語分割を取り上げ，上記課題への取り組みの１つとして，タグ付きコーパスとタグなしコーパスを相補的に利用することを試みる．具体的には，まず，日本語単語分割を入力文の文字間に単語境界を置くか置かないかの分類問題として定式化する．次に，タグ付きコーパスから作成した訓練データを用いて，決定リストＡを作成する．次に，タグなしコーパスを既存の形態素解析システムにより単語分割し，そこから誤り付き訓練データを作成する．そこから決定リストＢを作成する．そして決定リストＡのある順位の規則を default 規則ととらえ，それよりも下位の規則を省く．残された決定リストＡに決定リストＢを付加したものを新たな決定リストＣと見る．そして，決定リストＣの精度を訓練データとアダブーストの手法を利用して高める．実験の結果，誤り付きの訓練データを利用した効果と，ブースティングを行った効果が確認できた．

Use of tagged and untagged corpora for Japanese word segmentation

Hiroyuki Shinnou

Ibaraki University. Dept. of Systems Engineering

shinnou@dse.ibaraki.ac.jp

When we regard a natural language processing problem as a classification problem and solve it by an inductive learning method, the biggest problem is how to prepare correct training data. In this paper, we take up the Japanese word segmentation problem and propose the method to combine tagged and untagged corpora in the learning method to compensate for the shortage of training data. First, we regard the Japanese word segmentation problem as the classification problem to judge whether the word boundary exists between two characters or not. Next, we build the decision list 'dl.A' by training data made through a tagged corpus. In other hand, we make training data with some errors through an untagged corpus, and build the decision list 'dl.B' through it. We regard a rule in dl.A as the default rule and remove rules below it from dl.A. Next we attach dl.B to the left dl.A, and regard it as the decision list 'dl.C.' Finally, we improve the precision of dl.C by using Adaboost and correct training data. Through experiments, we confirmed effectiveness of boosting and use of the untagged corpus.

1 はじめに

自然言語処理の個々の問題を分類問題ととらえ，帰納学習の手法により問題解決を図る場合，最も重要な課題は，訓練データをどのように準備するかである．ここでは具体的な問題として日本語単語分割を取り上げ，上記課題への取り組みの1つとして，タグ付きコーパスとタグなしコーパスを相補的に利用することを試みる．

分類問題とはある事例 d がクラス $C_1, C_2, \cdots, C_n$ のうち，どのクラスに属するかを判定する問題である．自然言語処理の多くの問題は，この分類問題として定式化でき，分類問題に対する帰納学習の手法，例えば，決定木 [5]，決定リスト [7] あるいは最大エントロピー法 [6] などを利用することで問題解決が図れる．しかしこのような帰納学習の手法には，訓練データを準備するコストが高いという問題がある．訓練データとは，事例 d とその事例が属するクラス C_d（事例 d の正解）の組を多数集めたものである．自然言語処理の場合，事例は豊富にあるが，その事例の正解を人間が判断して付与する必要があり，このコストが高いことが大きな問題である．

この問題は帰納学習での事例の少なさの問題であり，事例の少なさの問題に対しては，背景知識を利用する手法 [4] や人手で作成したラフな既存知識を事例に融合させる手法 [9] が提案されている．また訓練データに正解を付与する必要がない教師なし学習も幾つか提案されている（例えば [8] など）．ただし教師なし学習では教師ありの学習以上には精度を上げることが難しい．一方，少数の正解付きの訓練データと多数の正解を付与されていない訓練データを相補的に利用する手法 [1] は，教師ありの学習から得られた規則を正解の付いていない訓練データを利用して改善していると見なすことができ，このアプローチは現実的に有望である．本論文ではこのアプローチをとる．

本論文の着眼点はタグなしコーパスから学習された規則は，一般には低いが，コーパスの量が多くなると，タグ付きコーパスから得られた精度の低い規則よりも，少しはましという点である．つまりタグ付きコーパスから得られた精度の低い規則の代りにタグなしコーパスから得られた規則を用いるアプローチが考えられる．しかも，このような規則の全体はタグ付きコーパスから得られた規則と見なすことができる．そのためブースティングの手法を利用して，この規則の精度を向上させることができる．

ここでは具体的な問題として日本語単語分割を考える．まず，日本語単語分割を入力文の文字間に単語境界を置くか置かないかの分類問題として定式化する．次にこの分類問題を決定リストを利用することで解決する．つまりタグ付きコーパスから作成した訓練データを用いて，決定リスト A を作成する．次に，タグなしコーパスを既存の形態素解析システムにより単語分割する．そこから誤り付き訓練データを作成し，そこから決定リスト B を作成する．そして決定リスト A のある順位の規則を default 規則ととらえ，それよりも下位の規則を省く．残った決定リスト A に決定リスト B を付加したものを新たな決定リスト C と見る．そして，決定リスト C の精度を訓練データとブースティングの1手法であるアダブースト [3] を利用して高める．

以下2章にタグ付きコーパスから決定リストを作成し，その決定リストを利用した日本語単語分割を述べる．3章では上記で説明したタグなしコーパスの利用方法を更に詳しく述べる．4章でアダブーストの利用について説明し，5章で実験，6章で考察，そして7章で結論を述べる．

2 決定リストによる日本語単語分割

日本語単語分割は入力文 ($s = c_1 c_2 \cdots c_n$) の各文字の間 (c_i と c_{i+1} の間の地点 b_i) に単語境界を置く（クラス $+1$）か置かない（クラス -1）かの分類問題としてとらえることができる．

分類問題は帰納学習の手法を利用して解決できる．帰納学習には様々な手法があるが，どの手法が優れているかは問題に依存するので，一概には言えない．ここでは決定リスト [7] を利用する．

2.1 決定リストの構築

決定リストの分類規則は証拠とクラスの組の順序付きの表である．ここで証拠とは属性とその属性の値の組である．実際の判別はリストの上位のものから順に，その証拠があるかどうかを調べ，その証拠があれば，それに対応するクラスを出力する．

決定リストの作成は概ね以下の手順による．

step 1 属性を設定する．

例えば n 個の属性を $att_1, att_2, \cdots, att_n$ とする。

step 2 訓練データから証拠とクラスの組の頻度を調べる．

訓練データ中のあるデータの属性 att の値が a であるとし，そのデータのクラスが C だとする．その場合，(att, a) という証拠とクラス C の組 $((att, a), C)$ の頻度に 1 を足す．これを訓練データ中の全データに対する全属性について行う．

step 3 証拠の判別力と分類クラスを導く．

$((att, a), C)$ の頻度が f_C であった場合，f_C の最大値を与える $\hat{C}$ が証拠 (att, a) に対する分類クラスとなる．またそのときの判別力 $pw(att, a)$ は以下で定義される．

$$pw((att, a)) = \log \frac{f_{\hat{C}}}{\sum_{C \neq \hat{C}} f_C}$$

step 4 判別力の順に並べる．

全ての証拠と分類クラスの組を判別力の大きい順に並べる．これによって作成できた表が決定リストである．

2.2 属性の設定

各文字間 b_i がどのクラスに属するかを判断する材料が属性である．本論文では b_i の属性として，表 1 の 7 種類を用意した．

表 1: 設定した属性

属性	値	
att_1	文字列	$c_{i-1}c_ic_{i+1}$
att_2	文字列	$c_ic_{i+1}c_{i+2}$
att_3	文字列	$c_{i-1}c_i$
att_4	文字列	c_ic_{i+1}
att_5	文字列	$c_{i+1}c_{i+2}$
att_6	字種の接続関係 1	$((c_i$の大分類字種), (c_{i+1}の大分類字種))
att_7	字種の接続関係 2	$((c_i$の細分類字種), (c_{i+1}の細分類字種))

6, 7 番目の属性として，字種の情報を利用している．ここでは字種を大分類と細分類の二つの観点から分類した．字種の大分類は 6 番目の属性，字種の細分類は 7 番目の属性で利用した．

字種の大分類は表 2 に示した 9 種類である．字種の細分類は大分類の平仮名の部分をその文字自身にしたものである．

表 2: 大分類字種

字種	意味	例
平	平仮名	あ，い，う，…
カ	カタカナ	ア，イ，ウ，…
数	漢数字	一，二，…，百，千，…
漢	漢字	亜，位，卯，…
N	英数字	0，1，2，…
ア	アルファベット	A，B，C，…
記	記号	、，。，「，…
○	小丸かぜロ	○
○	大丸かぜロ	○

また注意として，本論文の決定リストでは $default$ の証拠を導入していない．決定リストでは通常 $default$ という証拠を設けて，それ以下の判別力の証拠は表には入れない．$default$ は文脈上の証拠が決定リストに存在しない場合の処理ととらえられるが，ここでは大分類の字種の情報が必ずヒットするので，$default$ の証拠を含める必要がない．6 番目の属性からの証拠の最下位のものが，決定リストの最下位の証拠となる．

2.3 利用例

決定リストの利用例を示す．例えば「太郎は海でアイスクリームを食べた。」という入力文の 5 番目の文字 “で” と 6 番目の文字 “ア” の間にクラス +1 あるいは −1 を与えてみる．問題の地点が持つ証拠は以下の 7 種である．

$(att_1$,"海でア"), $(att_2$,"でアイ"), $(att_3$,"海で"),
$(att_4$,"でア"), $(att_5$,"アイ"), $(att_6$,"平カ"),
$(att_7$,"でカ")

後述する実験で得られた決定リスト A を用いると，各証拠の分類クラスと判別力は表 3 の通りである．

表 3 の中で “–” の記号のものは，決定リスト中にその証拠がないことを表す．また本来ならば，決定リスト中の順位を求めなければならないが，ここでは相対的な順位関係だけが必要であり，順位の値自体は必要でない．判別力の最も大きなものが

表 3: クラス判別の例

証拠	分類クラス	判別力
$(att_1$,"海でア")	−	−
$(att_2$,"でアイ")	−	−
$(att_3$,"海で")	+1	2.74377
$(att_4$,"でア")	+1	5.83188
$(att_5$,"アイ")	+1	1.64565
$(att_6$,"平カ")	+1	6.33293
$(att_7$,"でカ")	+1	8.64488

最上位の順位になるはずである．この場合，証拠 $(att_7$,"でカ") が最も大きな判別力を持つので，この証拠の分類クラス +1 が判定結果となる．つまり 5 番目の文字 “で” と 6 番目の文字 “ア” の間には単語境界を置くと判定する．

3 タグなしコーパスの利用

今，生のコーパスから既存のシステムを利用して自動的にタグをつける．こうして作られたタグ付きコーパスから訓練データを作成する．通常，この訓練データには誤りが含まれるために，直接は訓練データとして利用できない．この訓練データをここでは誤り付き訓練データと呼び，誤りのない一般の訓練データを単に訓練データと呼ぶ．訓練データから作成された分類器 A の精度と，誤り付き訓練データから作成された分類器 B の精度を比較すると，分類器 A の精度の方が分類器 B の精度よりも高い．しかし生のコーパスの量が大きく，しかも既存のシステムの精度がある程度以上ある場合，分類器 B の精度が分類器 A が内在する精度の低い規則よりも良いという状況が生じる．そこで分類器 A を使ってクラス判別を行うときに，精度の低い規則で判断する場合には，その代りに分類器 B により判別を行うことを試みる．

具体的には，日本語単語分割において，訓練データから決定リスト A を作成し，その決定リスト A のある順位以下の証拠を決定リスト A では精度の低い規則として，リストから取り除く．次に誤り付き訓練データから学習された決定リスト B を決定リスト A に付加して，全体として一つの決定リスト C を作成する (図 1 参照)．

本論文では決定リスト A から省かれる規則を判別力が 2.944 以下のものとした．この値は 0.95 の確率で正しい判断を行う場合の判別力

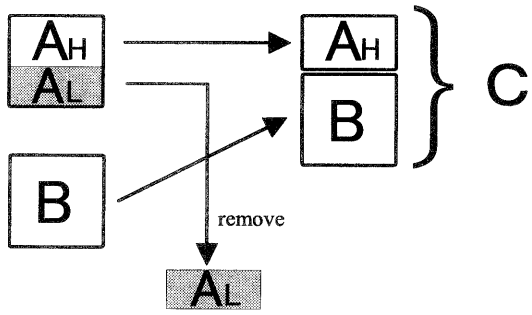

図 1: 決定リストの合成

$(\log(0.95/(1-0.95)) = 2.944)$ の値である．

4 アダブーストの利用

訓練データと誤り付き訓練データを単純に総和して，そこから決定リスト D を作った場合，決定リスト D の精度は決定リスト C を上回ることが予想される．それにも関わらず，決定リスト C を作成するには意味がある．決定リスト C において，決定リスト B の部分は一種の default 規則と見なせる．そのため，決定リスト C は訓練データのみから作成された分類器と見なせる．

誤りのない訓練データから作成された分類器に対してはブースティングにより，その精度を高めることができる．つまり決定リスト C の精度はブースティングにより，決定リスト D の精度を上回ることが期待できる．

ここではブースティングとして，アダブーストを利用する．アダブーストはブースティング方式の 1 つであり，現在まで多くの理論的検証と実験的実証から有効性が示されている．

アダブーストのアルゴリズムを図 2 [3] に示す．分類クラス (図 2 の Y) をここでは $\{+1, -1\}$ の 2 値とする．また訓練データを $(x_1, y_1), (x_2, y_2), \cdots, (x_m, y_m)$ で表す．ここで各 x_i はデータを表し，y_i はデータ x_i のクラスである．具体的に y_i は $+1$ あるいは -1 の値である．この訓練データに対して，分類問題に対する学習アルゴリズム，例えば，決定木や決定リストなどを適用して，分類器 h_1 を学習する．得られた分類器 h_1 を訓練データに適用すると，h_1 によって各 x_i の判

Given: $(x_1, y_1), \cdots, (x_m, y_m)$ where $x_i \in X, y_i \in Y = \{1, -1\}$
Initialize $Di(i) = 1/m$
For $t = 1, \cdots, T$

- Train weak learner using distribution D_t
- Get weak hypothesis $h_t : X \rightarrow Y$ with error

$$\epsilon_t = Pr_{i \sim D_t}[h_t(x_i) \neq y_i]$$

- Choose $\alpha_t = \frac{1}{2} ln(\frac{1-\epsilon_t}{\epsilon_t})$
- Update:

$$D_{t+1}(i) = \frac{D_t(i)}{Z_t} \times \begin{cases} e^{-\alpha_t} & if \quad h_t(x_i) = y_i \\ e^{\alpha_t} & if \quad h_t(x_i) \neq y_i \end{cases}$$

where Z_t is a normalization factor

Output the final hypothesis:

$$H(x) = sign(\sum_{t=1}^{T} \alpha_t h_t(x))$$

図 2: アダブースト

定クラスが得られる．今，x_i の実際のクラス y_i は与えられているので，分類器 h_1 が各 x_i に対して正しい判定を行ったかどうかを調べられる．これによって不正解のデータを集め，それら不正解のデータに対してある重みを付加して，訓練データ $(x_1, y_1), (x_2, y_2), \cdots, (x_m, y_m)$ を再構成する．そしてこの再構成された訓練データに対して，再び学習アルゴリズムを適用して，分類器 h_2 を学習する．これを T 回繰り返す．この繰り返しによって，T 組の分類器 $h_1, h_2, \cdots, h_T$ が得られる．実際の判定は入力データに対して各分類器が出力するクラスの多数決により行われる．

アダブーストのポイントは不正解のデータに課す重みの与え方である．概略，得られた分類器の誤り確率（図 2 における ϵ_t）が小さいほど重みが大きくなるように設定している．

本論文では．分類問題に対する学習アルゴリズムを決定リストに設定する．不正解データに与える重みをどのように反映させるかが問題である．ここでは，重みを頻度として与えることにした．例えば，「太郎が東京へ行く。」という文に以下のように単語境界 “/” が置かれたものが訓練データである．

太郎/が/東京/へ/行く/。

今，4 番目の文字 “東” と 5 番目の文字 “京” の間，つまり b_4 に対する証拠は以下の通りである．

$(att_1,$”が東京”$)$, $(att_2,$”東京へ”$)$, $(att_3,$”が東”$)$,
$(att_4,$”東京”$)$, $(att_5,$”京へ”$)$, $(att_6,$”漢漢”$)$,
$(att_7,$”漢漢”$)$

“東” と “京” の間には，単語境界がないので，クラスは -1 である．そして，決定リスト作成の step 2 で示したように，以下の証拠の頻度に 1 が足される．

$((att_1,$”が東京”$), -1)$, $((att_2,$”東京へ”$), -1)$,
$((att_3,$”が東”$), -1)$, $((att_4,$”東京”$), -1)$,
$((att_5,$”京へ”$), -1)$, $((att_6,$”漢漢”$), -1)$,
$((att_7,$”漢漢”$), -1)$

この頻度に加算される 1 という数値に重みを反映させる．

例えば，決定リスト h_k により上記例文の 4 番目の文字 “東” と 5 番目の文字 “京” の間の判定クラスが +1 と判定された場合，この判定は不正解である．そこで次の決定リスト h_{k+1} を作成するときに，上記の 7 つの各証拠の頻度に 1 ではなく，重み自身を加える．

つまり決定リストを作成する際には各訓練データには重みがついているとして，その重みが決定リスト作成の step 2 で各証拠と正解の組に付加する数値とする．図 2 のアルゴリズムでは正規化するために重みの総和が 1 になっているが，ここでは重みの最小値が 1 となるように正規化して計算を簡単にした．このため最初の決定リストを作成する際の各訓練データの重みは 1 であり，2 回目では正解のデータの重みは 1 で変化せず，不正解の部分の重みが大きくなる．

5 実験

5.1 タグなしコーパスの利用の効果

タグなしコーパスを利用する有効性を示すために，以下の 4 つの決定リスト A,B,C,D を作成し単語分割を行い，その精度を測った．

A 訓練データから作られた決定リスト A.

B 誤り付き訓練データから作られた決定リスト B.

C 訓練データと誤り付き訓練データからそれぞれ決定リストを作成し，前者の default 規則として後者を用いた決定リスト C（本手法）.

D 訓練データと誤り付き訓練データを単純にマージしたデータから作られた決定リスト D.

各実験で利用した訓練データ，誤り付き訓練データおよびテストデータは共通である．訓練データとしては，京大コーパス (約 4 万文) を利用した．京大コーパスは人手でタグをつけたコーパスであり，そこから正解付きの訓練データを作成できる．京大コーパスの中から 950117.KNP というファイルに納められた 1,234 文[1]をテストデータとした．結果，訓練データは京大コーパスからテストデータを除いた 35,717 文 である．テストデータ 1,234 文の中には，単語境界を置くか置かないかを判定する位置が 56,411 個所存在する．この 56,411 個所に対して正しいクラスを付与できた割合を正解率とする．次に毎日新聞 ’94 年度版[2]からランダムに 76,541 文取りだし，JUMAN 3.5 により形態素解析を行った．この 76,541 文の中には，単語境界を置くか置かないかを判定する位置が 3,242,058 個所存在し，JUMAN による形態素解析結果から，それらの各個所にクラスを付与できる．このクラスは誤りかも知れないが，JUMAN の正解率はかなり高いために，ほとんどのデータには正しいクラスが付与されていると考えられる．そこで本手法の評価が公平になるように，10 個所に 1 個 JUMAN により付与されたクラスを反転し，強制的に誤りの数を増やした．このデータを誤り付き訓練データとする．誤り付き訓練データの量は訓練データのほぼ 2 倍であり，混在する誤りの割合は 10% 強である．

決定リスト A,B,C,D の精度は以下の通りであった．

	A	B	C	D
誤り数	1,396	1,400	1,165	1,076
正解率	97.52%	97.51%	97.93%	98.09%

表 4: 各決定リストの精度

タグ付きコーパスだけを用いた決定リスト A よりもタグなしのコーパスを併用した決定リスト C や D の方が精度が高いことがわかる．

5.2 アダブーストの効果

先の決定リスト A,B,C,D をそれぞれの訓練データを利用してブーストさせた．決定リスト B, D はブーストさせると，精度が急激に下がった．一方，決定リスト A,C の精度は図 3 に示すように向上した．図 3 の縦軸は正解率，横軸はブーストティングの繰り返しの回数を表す．

決定リスト A も C も 3 つの決定リストを作成し，それらの多数決による判別の場合が，最もよい精度を出した．特に決定リスト C をブースティングさせて得られた精度 98.63% は，決定リスト D の最も良い精度 98.09% を越えており，本手法の有効性が示された．

[1]ここではコーパス中の記号 EOS の数を文の数としている．句点 “。” の数ではないことを注意しておく．

[2]京大コーパスは，毎日新聞 ’95 年度版からの作成されたおり，ここでの重複はない．

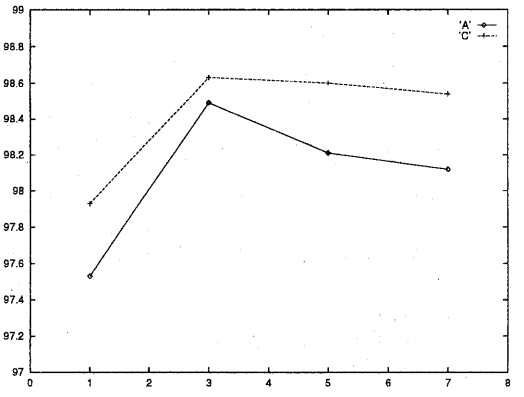

図 3: ブースティングの効果

6 考察

本論文ではタグなしコーパスを利用する手法を，単語分割問題に適用したが，基本的にはどのような分類問題に対しても適用可能である．ただし誤り付き訓練データを作成する適切な既存システムが存在することが本手法が有効に働くかどうかの鍵を握っている．

利用する既存システムは2つの要件を満たす必要がある．1つは正解率がある程度高いことである．本手法の枠組みから考えて，誤り付き訓練データから作成できる分類器の精度は，訓練データから作成できる分類器から取り除かれる規則群の精度以上であることが要求される．利用する既存システムは，この条件を満たすような誤り付き訓練データを作成できる必要がある．これは利用するタグなしコーパスの量とも関係するだろう．本実験で利用した既存システムの精度は90%弱，タグなしコーパスの量はタグ付きコーパスの量の約2倍となっている．

もう1つの要件は利用する既存システムが，学習で利用する属性とは別種の観点から作成されたものであることである．訓練データから得られた規則Aを誤り付き訓練データから得られた規則Bが，サポートできるのは，規則Bが規則Aとは別観点で得られた規則であるためだと思われる．規則Aでは判定が難しい問題も，別観点から得られた規則Bでは判定が容易であることがある．このような部分で規則Bが規則Aをサポートできるのだと考えられる．本論文で行った単語分割の場合，訓練データからの学習で利用する属性は，n-gramと字種である．誤り付き訓練データはJUMANの単語分割結果から得られている．これは辞書の情報を利用している．連語や1語として扱う複合語の問題があるため，辞書の情報は訓練データからだけでは100%得られることはない．このような辞書の情報はn-gramや字種とは別種の観点であるため精度向上につながっていると思われる．

既存システムが満たすべき要件，また利用するタグなしコーパスの量については，更なる調査が必要である．この点は今後の課題とする．

既存システムとして，学習される分類器自身を利用するアプローチも考えられる．ただしこの場合，2種類の観点を利用していることにはならなず，なんらかの工夫が必要である．Blum と Mitchell は，ラベル付きの訓練データから学習させた規則の精度を，ラベルなしの訓練データを利用して高める Co-Training という手法 [1] を提案しているが，そこでも学習が2種類の観点を利用しているという点がポイントになっている．Co-Training の手法を利用して， Collins と Singer は少数の規則群と大量のラベルなしの訓練データから固有表現抽出規則を学習した [2]．そこでも2種類の観点を利用することがポイントになっている．固有表現の場合，表記の情報と文脈の情報が2種類の観点に対応する．

最後に本手法を単語分割問題の手法と見た場合の特徴を述べておく．日本語単語分割は JUMAN のような既存のシステムでも既に十分高精度であり，実用レベルに達している．残された課題は未知語の検出である．未知語の問題への1つの対処方法として，文字ベースの単語分割手法を用いることがあげられる [10, 11]．文字ベースの手法では，未知語という概念自身がないので，未知語の問題を受けない．ここで行った単語分割の方法も文字ベースの手法の一種であり，未知語の問題を回避できている．どの程度未知語の検出能力があるかを，決定リスト A について調べた．まずテストデータに含まれる単語の中から，訓練データに出現する単語を除いた．その結果 832 種類の単語が残った．これが決定リスト A における未知語と考えられる．このうちブースティング後の決定リスト A が認識できた単語は 562 種類であり，67.5% の再現率があった．これは本手法のように単語分割を決定リストとブースティングから行うアプローチが，未知語に対して，ある程度，有効に働いていることを示している．

7 おわりに

本論文では，自然言語処理の個々の問題を分類問題として定式化して解く場合に，訓練データを用意するコストが高いという問題に対して，タグ付きコーパスとタグなしコーパスを相補的に利用する手法を提案した．そこでは，タグなしコーパスから既存のシステムによって誤り付きの訓練データを作成し，そこから得られる規則を，通常の訓練データから作成した分類器の精度の低い規則の部分と取り代える．このように作成された分類器をアダブーストの手法を利用することで精度を高める．

ここでは具体的な問題として日本語単語分割を取り上げ，学習アルゴリズムとして決定リスト，既存のシステムとして JUMAN を利用した．実験の結果，タグなしコーパスを利用した効果およびアダブーストによる効果を確認できた．

既存システムが満たすべき要件，また利用するタグなしコーパスの量についての調査を今後の課題とする．

参考文献

[1] Avrim Blum and Tom Mitchell. Combining Labeled and Unlabeled Data with Co-Training. In *11th Annual Conference on Computational Learning Theory (COLT-98)*, pp. 92–100, 1998.

[2] Michael Collins and Yoram Singer. Unsupervised Models for Named Entity Classification. In *1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pp. 100–110, 1999.

[3] Yoav Freund, Robert Schapire (訳: 安倍直樹). ブースティング入門. 人工知能学会誌, Vol. 14, No. 5, pp. 771–780, 1999.

[4] Takefumi Yamazaki Hussein Almuallim, Yasuhiro Akiba and Shigeo Kaneda. Induction of Japanese-English Translation Rules from Ambiguous Examples and a Large Semantic Hierarchy. 人工知能学会誌, Vol. 9, No. 5, pp. 730–740, 1994.

[5] J. Ross Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publisher, 1993.

[6] Adwait Ratnaparkhi. Maximum Entropy Models for Natural Language Ambiguity Resolution. In *PhD thesis*. University of Pennsylvania, 1998.

[7] David Yarowsky. Decision Lists for Lexical Ambiguity Resolution: Application to Accent Restoration in Spanish and French. In *32th Annual Meeting of the Association for Computational Linguistics*, pp. 88–95, 1994.

[8] David Yarowsky. Unsupervised Word Sense Disambiguation Rivaling Supervised Methods. In *33th Annual Meeting of the Association for Computational Linguistics*, pp. 189–196, 1995.

[9] 秋葉泰弘, 石井恵, フセイン・アルモアリム, 金田重郎. 人手作成ルールと事例に基づく英語動詞選択ルールの学習. 自然言語処理, Vol. 3, No. 3, pp. 53–68, 1996.

[10] 小田裕樹, 北研二. PPM^* モデルによる日本語単語分割. 情報処理学会自然言語処理研究会, NL-128-2, 1998.

[11] 小田裕樹, 森信介, 北研二. 文字クラスモデルによる日本語単語分割. 自然言語処理, Vol. 6, No. 7, pp. 93–108, 1999.