異種の言語知識を利用した品詞判定知識の自動獲得

平川秀樹　吉村裕美子　小野顕司
(株)東芝　研究開発センター

1　はじめに

近年、タグ付けコーパスを利用して様々な言語処理のための知識を自動獲得するための研究が盛んである。英語の品詞タグ付けでは、統計手法およびそれに文法知識を組み合わせたものなど数種の手法が報告されているが([1]-[5]) その精度がトレーニングコーパス（あらかじめ正解をタグ付けされたデータセット）の質と量に依存することが共通の問題としてある。一方、自然言語処理技術の発展に伴い、高度な知識を蓄えた実システムが開発されるようになってきた。このようなシステムではすでに核となる知識を搭載しているため、精度の向上のためには、語に依存した個別知識の獲得が必要である。このためには非常に大きなトレーニングコーパスが必要になり、タグ付けされたコーパスを作成・利用できる規模を超えている。そこで我々は、プレーンテキストコーパスを知識獲得の源とするアプローチに視点を転換し、既存システムの各モジュールに蓄えらえる信頼度の高い異種の言語知識源を互いに協調的に作用させることでタグ付けコーパスに代える、言語知識の自動獲得手法を考案した。本稿では、特に、当社機械翻訳システムの品詞判定モジュールと構文解析モジュールを利用して、既存品詞判定知識を自動修正する知識の獲得について述べる。

2　品詞判定規則自動獲得システムの概要

図1に、品詞判定規則を自動獲得するシステムの概要を示す。当システムの中心となるのが文解析器であり、品詞判定モジュールと構文解析モジュールを有する。両モジュールは、それぞれ既存の品詞判定規則、構文解析規則を持っている。

品詞判定モジュールは、仮説の生成器として働き、コーパス中の個々の文に対して優先度付きの品詞列候補を複数生成する。ここで最も優先度の高い品詞列（第1品詞列）が、品詞判定モジュールが正解としているものである。一方、構文解析モジュールは仮説に対する検証器として働き、この優先度順に個々の品詞列を解析し構文解析可能か否かを判定し、解析可能な品詞列を得るか、品詞列候補が尽きたところで処理を終える*。

*品詞の組合せ爆発を回避するため、生成する品詞列候補数は所定数に制限している。

次に、両モジュールの出力を受けて、3節で述べる方法で新しい品詞判定規則の候補を生成し、続いて、規則抽出モジュールにおいて、4節で述べる統計的な手法を用いて規則候補の有益性を検証し、新たな品詞判定規則を抽出する。

規則獲得後は、既存の品詞判定規則、構文解析規則と、獲得された品詞判定規則をプラスして文解析が行われ、さらなる規則獲得に利用される。

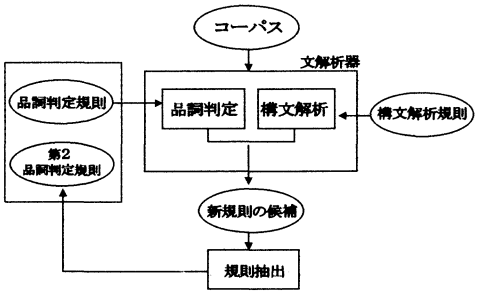

図 1: システム概要

3　品詞判定規則の候補生成

品詞判定規則の候補の生成は、品詞判定モジュールと構文解析モジュールの判定結果の食い違いより導かれる。構文解析に失敗した第1品詞列と最終的に構文解析に成功した品詞列との間の割り当て品詞の変化に注目することで、品詞判定誤りの語とその文脈（隣接語句）を検出できる。このような品詞の変更を文脈とともに、品詞を調整するための新たな品詞判定規則（第2品詞判定規則）の候補として蓄えていく。

図2に候補生成の例を示す。本方式では、着目語の前後2語の品詞を文脈としている。第2品詞判定規則のシンタックスは以下のように、「:」の前が品詞変更情報、後が文脈条件となっている。

単語 (変更前品詞) →単語 (変更後品詞):
前語 1- 前語 2-$- 後語 1- 後語 2

「$」は着目単語、「前語 1,2」「後語 1,2」はそれぞれ前後に隣接する語の品詞情報（ラベル）を示す。前置詞など機能語やbe, haveなど一部の重要語については1語に1つのラベルを割り当て、動詞の活用について

も別個のラベルとしている（ラベルの異なり数513†）。「rank(v) → rank(n): v-det-$-'in'-det」の場合は、単語rankの前に動詞「v」、冠詞「det」という品詞の並びが隣接し、後に前置詞'in'（引用符で囲ったものは単語自体をラベル化したもの）、「det」の品詞の並びが後続するときに、動詞「v」から名詞「n」に品詞の変更をすることを示す。

この候補生成を大量文に対して行い、次節に述べるようにその結果に対して統計的な検証を施すことによって第2品詞判定規則を作成する。

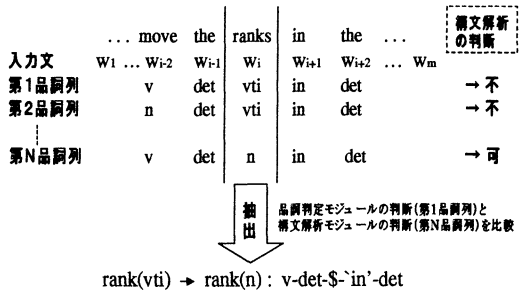

図 2: 規則候補の抽出

4 大規模コーパスを使った規則候補の検証

表1は、ある単語Wを含むコーパス中の文を解析し、特定の文脈条件E（=P1-P2-$-P3-P4）において、特定の単語Wに対する品詞の候補として品詞X、品詞Yの2種類がある場合に、品詞判定モジュールが第1品詞としてXを割り当てた後、最終的に構文解析が成功する過程における品詞の遷移を示す。Aは単語Wに対して第1品詞であるXとして構文解析が成功したケース、Bは第1品詞として割り当てられXでは構文解析を成功せず、Yに変更された結果、構文解析を成功したケースである。

Pos X	A
Pos X → Pos Y	B

表 1: 文脈条件Eにおける単語Wの解析過程での品詞遷移

A、Bそれぞれの文数をNa、Nbとする。ここで、文脈条件Eが単語Wの品詞を決定するのに適度な詳細度を持っており、かつ、構文解析モジュールは解析精度が高く、多くの文を対象とした場合には、誤った品詞列候補に対しては解析失敗の割合が多く、正しい品詞列候補に対しては解析成功の割合が多いと仮定する。この仮定の元では、品詞判定モジュールが文脈条件Eにおいて第1品詞として単語WにXを当てた時、Na+Nbに対するNbの割合が有意的に高くなるか低くなるかいずれかに偏るはずである。この数値を変更率と呼ぶこととし、以下のように立式され、文脈条件Eにおける第1品詞と最終的に構文解析に成功する品詞の遷移ごとに算出する。これは、生成する個々の規則ごとに算出することに他ならない。

$$\text{変更率}_{W,E}(X \rightarrow Y) = \frac{Nb}{Na + Nb}$$

上記仮定が正しければ、十分に大規模なコーパスに対して、規則候補を抽出した後、規則ごとにの変更率を算出すれば、正しい品詞変更を意図する規則は変更率の高いところに集中し、逆に、誤った品詞変更を意図する規則は変更率の低い側に集中することが予想される。よって、変更率をキーとした望ましい規則の選出が可能になると言える。

一定の出現頻度があり、変更率$_{W,E}$(Y → X)有意的に高い規則については、品詞判定モジュールの最初から所定文脈EではYではなくXを割り当てるように適用すれば、構文解析失敗・品詞変更に要する時間を丸ごと短縮できる。実際には、構文解析規則の不備などにより、誤った品詞列を誤った解析木として解析し、誤った規則を抽出することもあり得る。しかし、正しい品詞列を正しい解析木で解析するための知識が構文解析規則に不足している以上、新規獲得の誤った品詞判定規則を使っても使わなくても、誤った解析木が導かれることに変わりはなく、全体としての解析精度にマイナス効果は少ない‡。得られる結果が同じであっても、規則として適用すれば、前述のように処理時間を短縮できるという効果がある。このように、本方式は誤った知識の獲得に対してロバスト性が高い方式であると言える。

5 規則抽出実験

英語ニュース記事776,219文（約73MB）に対して前節で述べた方法で18,868種類の規則候補を収集した。規則候補の頻度と変更率を軸とした規則の分布の集中した部分を切り出し図3に示す。便宜上、規則数100を超えるものも100として表示している。低頻度の例から抽出された規則は信頼度が低いため、今回は頻度6以下の規則は対象から外すこととし、この段階で規則候補は212となった。

†このうち前置詞、群前置詞が410を占める。

‡文脈条件が不十分な場合においては、構文解析結果が悪化する場合もある。

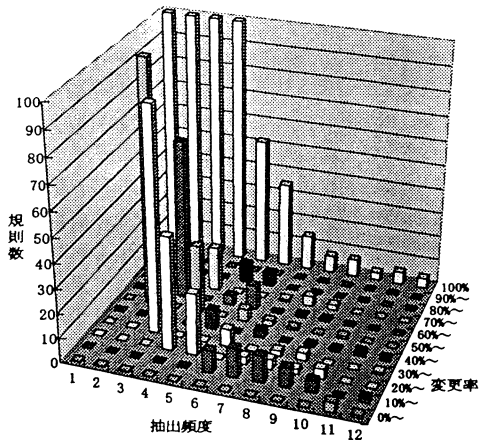

図 3: 規則候補の分布

変更率	計	正	(%)	誤	(%)	他
0 〜	56	3	(5)	42	(75)	11
10 〜	44	4	(9)	27	(61)	13
20 〜	13	4	(31)	7	(54)	2
30 〜	9	5	(56)	1	(11)	3
40 〜	11	5	(45)	2	(17)	4
50 〜	4	2	(50)	1	(25)	1
60 〜	5	4	(80)	0	(0)	1
70 〜	8	6	(75)	2	(25)	0
80 〜	0	0	(0)	0	(0)	0
90 〜	3	2	(66)	0	(0)	1
100	59	33	(56)	14	(24)	12

表 2: 規則の判定結果

抽出した規則が妥当なものであるかを評価するために、規則の抽出元となった実例文を参照し人手で品詞判定を行うことにより、抽出された規則の妥当性をチェックした。評価値は以下の3種とする。表2にその結果を示す。各数字は規則の数を表し、「正」「誤」のあとの「%」は合計数の中に占める割合を表す。

(1) 正 規則として妥当であり他の文に適用してよい。

(2) 誤 規則として妥当でない。構文解析規則不足のために誤った品詞列で構文解析が成功したもの。

(3) 他（どちらともいえない） 文脈に含めた単語数および品詞のレベルでは、一意には正誤をつけられないもの、およびどちらの品詞も正しくないもの。

表から明らかなように変更率30%を境にそれ以下では誤りの率が高く、逆にそれ以上では正しいものの割合が高い。特に60 〜 99%では誤り率は非常に低い。このことは、文脈条件が単語の品詞を決定するのに適度な詳細度を持っており、変更率を基準にすれば正しい知識の抽出ができることの裏付けとなっている。なお、変更率100%で誤りの規則の率が高まるのは、構文解析規則に不備のある定型的なパターンが実験に用いたコーパスに多用されていたためである。

本方式では、変更率の計算は人手によらず行うことができるので、第2品詞判定規則の抽出は自動的に行える。今回は、変更率60%以上を基準とした。この場合、表2より「正」45、「誤」16、「他」14となる。誤りが含まれているが、4節に述べた理由により、その悪影響はこの比率より少ないと予想できる。

得られた75規則の中には、「return call(n)」と「return(v)+call」のように合成語から単独語の連続からなる系列への変更を意図する規則が5つあった。このケースでは、returnを着目語、callを後語1として規則候補の生成を行ったため、合成語としての系列における文脈語は処理に組み込まれていない。人手による評価では「他」3、「正」2と明らかなマイナス評価とはなっていないが、今回はこの5規則の適用は控え、今後改めて合成語としての文脈語を組み込んで実験をしなおすこととした。これで、最終的に規則の数は70（正：43、誤：16、他：11）となった。

一方、0 〜 30%では誤りの割合が非常に高い。これは、解析規則の不足により正しい品詞列が正しく処理できず、誤った品詞列を導いたためである。これらについては、今後「優先しない品詞系列」という否定的知識としての利用が可能であるが、今回の実験には組み込んでいない。また、ここであげられた事例は構文解析規則の改善に役立てることができる。

6 獲得規則の適用

獲得した規則の効果を検証するため、機械翻訳システムに組み込んでニュース記事241,080文（4,620,046語、約26MB）を機械翻訳し、処理時間と訳文の変化を見ることにした。結果の信頼度を保証するため、テスト文書は規則適用時に用いたものと別個のものとした。用いた計算機はEWS SUN Ultra U2E/200である。

実験の結果、518文に対して第2品詞判定規則が適用された。当518文の解析処理時間を表3に示す。品詞判定処理は7%、構文解析処理は43%、全体処理時間は23%の時間短縮が見られた。また、解析を成功する文の割合は9%向上した。

	適用前	適用後	
品詞判定時間	85.84	80.43	(-5.41, -7%)
解析処理時間	471.62	269.86	(-202, -43%)
全体処理時間	787.63	606.98	(-180, -23%)
解析成功文率	70%	79%	

表 3: 処理時間（単位：秒）

次に、規則が適用された518文の内、翻訳結果に変化があった文を取り出し、訳文の精度の変化を調べた。訳文の評価結果と、悪化事例についてはその原因の分類を表4に示す。これから、改善率は $\frac{105-14}{518} = 18\%$ と概算できる。

変化のあった文数	127
改　善	105
悪　化	14
（原因内訳）	
(a) 構文解析知識の不備	11 （規則数: 4）
(b) 規則適用仕様の不備	3 （規則数: 2）
大差なし	8

表 4: 訳文における変化

以下に、改善事例を1例挙げる。新旧とも文全体として翻訳できているが、旧訳では、carryの品詞を動詞ととらえた無理な訳を導いているのに対し、新訳では正しく名詞として処理している。（「***」は単語がないこと、つまり直前の語が文末であることを示す。）

原文: Robert Smith rushed for 86 yards on 17 carries.

規則: carry(v) → carry(n):'on'-dig-$-punc-***

旧訳: １７の上で８６ヤードで急がせられたロバート・スミスは達する。

新訳: ロバート・スミスは１７キャリーで８６ヤード突進した。

悪化事例は、予想通り、規則中の「誤」の割合に比較して数は少ない。表中の(a)は、構文解析不備により獲得された変更率100%の規則によるもので、該当した文については、規則抽出時の文との微細な違いにより、学習前の品詞でも構文解析が成功する文、あるいは規則にしたがって品詞を変更しても構文解析が成功せず、変更前の品詞で部分的に翻訳を行った方が良いという文であった。人手による規則の振り分けを行わない自動獲得方式のため、少数ながらこのような副作用の介入が伴う。

(b)は規則を適用する際の文脈条件と原文との照合の仕様の不備によるものである。2件は、文脈条件中でbe、haveを他の動詞と区別しているのに原文との照合時には区別していないこと、1件は、多品詞語を文脈条件と照合するのに、品詞判定モジュールが優先していない品詞として照合を成功させていることが原因であった。いずれも仕様詳細化により容易に対処可能である。

今回の実験では、規則抽出を行ったコーパスサイズが小さいため獲得規則数が少なく、評価実験で規則が適用された文数も少ない。だが、低頻度のため削除した規則候補の質を評価してみたところ、「正」の範疇のものが多くみられ、コーパスサイズの増加により規則数を増加できる。今回の実験では、仮にコーパスサイズを2倍とすれば、規則数は4倍程度になると見積もれる。図3の分布が高頻度側にシフトした姿を考えれば、コーパスサイズの増加に従って獲得規則数は急激に上昇すると予測がつく。一方、コーパスサイズが大きくなれば、規則抽出時の文脈条件に含める語数を広げたり、品詞ラベルを細分類したり、ラベルを語彙レベルに落とす語の種類を増やすなどの詳細化も可能となる。

7 おわりに

品詞判定知識をプレーンテキストコーパスから自動獲得するための新しいアプローチを提案し、実験によりその有効性を数値的に確認することができた。本手法の最大の特長は、(1) 大規模なタグ付けコーパスが不要、(2) 誤った知識の獲得に対してロバスト性が高い、の2点である。知識をゼロから獲得するのではなく、相当の精度をもつ自然言語処理システムの精度をさらに向上するための知識を獲得しようという場合に特に適した手法である。

参考文献

[1] Brill, E. 1992: A simple Rule-Based part of Speech Tagger, in *Proceedings of Third Conference on Applied Natural Language Processing*, pp. 152-155.

[2] Brill, E. 1995: Unsupervised Learning of disambiguation rules for Part of Speech Tagging, *Workshop on Very Large Corpora.*

[3] Church, K. 1988: A Stochastic parts program and noun phrase parser for unrestricted text", in *Proceedings of Second Conference on Applied Natural Language Processing*, Austin, Texas, pp.126-143.

[4] Kupiec, J. 1992: Robust part-of-speech tagging using a Hidden Markov Model, *Computer Speech & Language*, 6(3), pp.225-242.

[5] Tapanainen, P. and Voutilainen, A. 1994. Tagging accurately - Don't guess if you know, In *Proceedings of the Fourth Conference on Applied Natural Language Processing* .