最大エントロピー法に基づくモデルを用いた日本語係り受け解析

内元清貴[†] 関根聡[††] 井佐原均[†]

本論文ではME（最大エントロピー法）に基づくモデルを利用した統計的日本語係り受け解析手法について述べる．一文全体の係り受け確率は，一文中のそれぞれの係り受けの確率の積から求められると仮定し，それぞれの係り受けの確率はMEによって学習した係り受け確率モデルから計算する．この確率モデルは，学習コーパスから得られる情報を基に，2つの文節が係り受け関係にあるか否かを予測するのに有効な素性を学習することによって得られる．我々が素性として利用する情報は，2つの文節あるいはその文節間に観測される情報，たとえば，文節中の表層文字列，品詞，活用形，括弧や句読点の有無，文節間距離およびそれらの組合せなどである．本論文では，我々が用いた素性のそれぞれを削除したときの実験結果を示し，どの素性がどの程度係り受け解析の精度向上に貢献するかについて考察する．また，学習コーパスの量と解析精度の関係についても考察する．我々の手法による係り受けの正解率は，一文全体の係り受けを文末から文頭へ向かって決定的に解析した場合，京大コーパスを使用した実験で87.1%と高い精度を示している．

Japanese Dependency Structure Analysis Based on Maximum Entropy Models

KIYOTAKA UCHIMOTO,[†] SATOSHI SEKINE[††] and HITOSHI ISAHARA[†]

This paper describes an analysis of the dependency structure in Japanese based on the maximum entropy models. Japanese dependency structure is usually represented by the relationships between phrasal units called bunsetsu. We assume that the overall dependencies in a sentence can be determined based on the product of the probabilities of all dependencies in a sentence. The probabilities of dependencies between bunsetsus are estimated by a statistical dependency model learned within a maximum entropy framework. This model can be created by learning the features that are useful for predicting the dependency between bunsetsus from the training corpus. We are using information about a bunsetsu itself as features, such as character strings, parts of speech, and inflection types. We are also using information between two bunsetsus as features, such as the existence of brackets or punctuation and the distance between bunsetsus. We compare the performance of our method with and without each feature and discuss the contribution of each feature. And we discuss the effect of the size of the training corpus on the performance of our method. The accuracy of our method for obtaining the dependency of bunsetsus is 87.1% using the Kyoto University corpus when we parse a sentence deterministically from its end to the beginning.

1. はじめに

係り受け解析は日本語解析の重要な基本技術の1つとして認識されている．一般に係り受け解析では2文節間の係りやすさを数値化した係り受け行列を作成し，動的計画法などを用いて一文全体が最適な係り受け関係になるようにする．この場合，問題は2文節間の係りやすさをどのように決めるかということと，どのようにして一文全体の係り受け関係を決定するかということである．

これまでルールベースの研究では，2文節間の係りやすさを決める規則を人間が作成していた．しかし，係り受け解析で有効だと考えられている素性数は多く，互いに競合することも多いため，それでは網羅性，一貫性という点で問題がある．さらに，2文節間の係りやすさは解析するテキストの種類に依存すると考えられるため，異なる種類のテキストを解析しようとすると規則を変更する必要が生じやすく，その変更作業にかかるコストも高い．そこで我々は，2文節の係り受けの確率を計算するためのモデルをコーパスから統計的に学習し，その確率の大小を係りやすさの目安とする手法[1)]を採用した．このような統計的な構文解析手

† 郵政省通信総合研究所
Communications Research Laboratory, M.P.T.
†† ニューヨーク大学
New York University

法については，英語，日本語など言語によらず，いろいろな提案が80年代から数多くあり，現在，英語についてはRatnaparkhiのME（最大エントロピー法）に基づく学習モデルを利用した解析[2]が，精度，速度の両方の点で最も進んでいる手法の1つと考えられている．また日本語についても，限られた係り受け現象に対してではあるが，MEを利用した解析が他の学習手法によるものよりも優れていると報告されている[3]．そこで我々もMEに基づく学習モデルを利用する．簡単に説明すると，MEは，素性が与えられたとき学習データ中に観測された素性の頻度などからそのデータに特徴的な素性を重み付けする仕組みのことである．素性とは，我々の場合，2つの文節間の係り受けの確率を計算するための情報であり，具体的には表層文字列，品詞，活用形，括弧や句読点の有無，文節間距離およびそれらの組合せなどを利用した．そして，テストの際には，学習されたモデルを基にテスト文中に与えられた2つの文節の素性からその2つの文節の係り受けの確率を計算する．複数の文節の組合せの確率は，それぞれの文節係り受けの確率の積を利用している．つまり，文の確率は，その文中にあるすべての係り受けの確率の積で求められる．

日本語の係り受けには，主に以下の特徴があるとされている．我々はこれらの特徴を仮定し，文末から文頭に向けて解析する手法を用いることによって一文全体の係り受け関係を決定する☆．

（1）後方を修飾する．
（2）係り受け関係は交差しない（非交差条件）．
（3）係り要素は受け要素を1つだけ持つ．
（4）ほとんどの場合，係り先決定には前方の文脈を必要としない．

これまでも，文末からの解析手法はルールベースの解析手法において利用されてきた[4]．ところが，一文全体としてどの係り受け関係から優先して決定していくかといった優先度を組み入れることが難しく，ヒューリスティックによる決定的な手法として利用せざるをえなかった．したがって，前方からの一般的な構文解析の手法に比べて精度の問題が指摘されていた．しかし，文末から解析を行うという手法を統計的解析に結び付けることにより解析速度を落とすことなく，精度に関する指摘を解決することができる．

☆（4）の特徴はあまり議論されてはいないが，我々が行った人間に対する実験で90%以上の割合で成立することが確認された．

2. 確率モデルの学習

この章では2文節間の係り受け確率を計算するためのモデルをコーパスから統計的に学習する方法について述べる．係り受け確率モデルとしてME（最大エントロピー法）に基づくモデルを採用し，学習コーパスから，2文節の係り受けとそこで観測される素性との依存関係を学習することによって係り受け確率を計算する．

まず，MEについて基本的な説明をし，その後，それを利用した2文節間の係り受け確率を計算するためのモデルを説明する．

2.1 ME（最大エントロピー法）に基づくモデル

一般に確率モデルでは，文脈（観測される情報のこと）とそのときに得られる出力値との関係は既知のデータから推定される確率分布によって表される．いろいろな状況に対してできるだけ正確に出力値を予測するためには文脈を細かくする必要があるが，細かくしすぎると既知のデータにおいてそれぞれの文脈に対応する事例の数が少なくなりデータスパースネスの問題が生じる．MEモデルでは，文脈は素性と呼ばれる個々の要素によって表され，確率分布は素性を引数とした関数として表される．そして，各々の素性はトレーニングデータにおける確率分布のエントロピーが最大になるように重み付けされる．このエントロピーを最大にするという操作によって，既知データに観測されなかったような素性あるいはまれにしか観測されなかった素性については，それぞれの出力値に対して確率値が等確率になるようにあるいは近付くように重み付けされる．このためMEモデルはデータスパースネスに強いとされている．このモデルは，たとえば言語現象などのように既知データにすべての現象が現れえないような現象を扱うのに適したモデルであるといえる．

以上のような性質を持つMEモデルでは，確率分布の式は以下のように求められる．文脈 b $(\in B)$ で出力値 a $(\in A)$ となる事象 (a,b) の確率分布 $p(a,b)$ をMEにより推定することを考える．文脈 b は k 個の素性 f_j $(1 \leq j \leq k)$ の集合で表す．そして，文脈 b において，素性 f_j が観測されかつ出力値が a となるときに1を返す以下のような関数を定義する．

$$g_j(a,b) = \begin{cases} 1 \ (exist(b,f_j) = 1 \ \& \ \text{出力値} = a) \\ 0 \ (\text{それ以外}) \end{cases} \quad (1)$$

これを素性関数と呼ぶ．ここで，$exist(b,f_j)$ は，文脈 b において素性 f_j が観測されるか否かによって1あるいは0の値を返す関数とする．

次に，それぞれの素性が既知のデータ中に現れた割合は未知のデータも含む全データ中においても変わらないとする制約を加える．つまり，推定するべき確率分布 $p(a|b)$ による素性 f_j の期待値と，既知データにおける確率分布 $\tilde{p}(a,b)$ による素性 f_j の期待値が等しいと仮定する．これは以下の制約式で表せる．

$$\sum_{a\in A,b\in B} \tilde{p}(b)p(a|b)g_j(a,b) = \sum_{a\in A,b\in B} \tilde{p}(a,b)g_j(a,b) \quad for\ \forall f_j\ (1\le j\le k) \qquad (2)$$

ここで，$\tilde{p}(b)$，$\tilde{p}(a,b)$ は，$freq(b)$，$freq(a,b)$ をそれぞれ既知データにおける事象 b，(a,b) の出現頻度として以下のように推定する．

$$\tilde{p}(b) = \frac{freq(b)}{\sum_{b\in B} freq(b)} \qquad (3)$$

$$\tilde{p}(a,b) = \frac{freq(a,b)}{\sum_{a\in A,b\in B} freq(a,b)} \qquad (4)$$

次に，式 (2) の制約を満たす確率分布 $p(a,b)$ のうち，エントロピー

$$H(p) = -\sum_{a\in A,b\in B} \tilde{p}(b)p(a|b)\ log\,(p(a,b)) \qquad (5)$$

を最大にする確率分布を推定するべき確率分布とする．これは，最も一様な分布となる．このような確率分布は唯一存在し，以下の確率分布 p^* として記述される．

$$p^*(a|b) = \frac{\prod_{j=1}^{k} \alpha_{a,j}^{g_j(a,b)}}{\sum_{a\in A}\prod_{j=1}^{k} \alpha_{a,j}^{g_j(a,b)}} \quad (0\le \alpha_{a,j}\le\infty) \qquad (6)$$

ただし，

$$\alpha_{a,j} = e^{\lambda_{a,j}} \qquad (7)$$

であり，$\lambda_{a,j}$ は素性関数 $g_j(a,b)$ のパラメータである．このパラメータは文脈 b のもとで出力値 a となることを予測するのに素性 f_j がどれだけ重要な役割を果たすかを表している．訓練集合が与えられたとき，パラメータの推定には Improved Iterative Scaling (IIS) アルゴリズム[5]などが用いられる．

2.2 係り受け確率モデル

本節では前節で述べた ME に基づくモデルを用いて，2 つの文節が係り受け関係にある確率を計算する方法について述べる．出力値 a を 2 文節が係り受け関係にあるか否かの 1，0 の二値とし，k 個の素性 f_j $(1\le j\le k)$ を考えるとき，文脈 b における 2 文節の係り受け確率 $p^*(1|b)$ は式 (6) を用いて以下のように求められる．

表 1 文脈，出力値，素性関数値の関係
Table 1 Relationship among history, output value, and feature function value.

$g_1(a,<F_1,F_2>)$	値	$g_2(a,<F_1,F_2>)$	値
$g_1(1,<0,0>)$	0	$g_2(1,<0,0>)$	0
$g_1(0,<0,0>)$	0	$g_2(0,<0,0>)$	0
$g_1(1,<0,1>)$	0	$g_2(1,<0,1>)$	1
$g_1(0,<0,1>)$	0	$g_2(0,<0,1>)$	1
$g_1(1,<1,0>)$	1	$g_2(1,<1,0>)$	0
$g_1(0,<1,0>)$	1	$g_2(0,<1,0>)$	0
$g_1(1,<1,1>)$	1	$g_2(1,<1,1>)$	1
$g_1(0,<1,1>)$	1	$g_2(0,<1,1>)$	1

$$p^*(1|b) = \frac{\prod_{j=1}^{k} \alpha_{1,j}^{g_j(1,b)}}{\prod_{j=1}^{k} \alpha_{1,j}^{g_j(1,b)} + \prod_{j=1}^{k} \alpha_{0,j}^{g_j(0,b)}} \qquad (8)$$

ここで，2 つの文節が係り受け関係にある確率の計算方法を簡単に説明する．たとえば，f_1，f_2 の 2 つの素性を考えるとき，このモデルでは各素性が観測されるか否かを 0 か 1 で表現したベクトルを用いて，$\{<F_1,F_2>\,|\,<0,0>,<0,1>,<1,0>,<1,1>\}$ の 4 種類の文脈が扱える．各文脈，出力値に対してそれぞれ式 (1) のように素性関数 $g_j(a,<F_1,F_2>)$ を定義すると，**表 1** のように素性関数値が決まる．学習コーパスに現れる事象の確率分布 $p(a,<F_1,F_2>)$ から式 (7) で表されるパラメータを推定し，$\alpha_{0,1}$，$\alpha_{0,2}$，$\alpha_{1,1}$，$\alpha_{1,2}$ を得たとする．テストコーパス中の任意の 2 文節の係り受け確率 $p^*(1|<F_1,F_2>)$ は，各素性関数値（表 1）を式 (8) に代入することによって計算できる．たとえば，文脈 $<1,0>$ に対しては次のように確率が計算される．

$$\begin{aligned} &p^*(1|<1,0>) \\ &= \frac{\alpha_{1,1}^{g_1(1,<1,0>)}\alpha_{1,2}^{g_2(1,<1,0>)}}{\alpha_{1,1}^{g_1(1,<1,0>)}\alpha_{1,2}^{g_2(1,<1,0>)} + \alpha_{0,1}^{g_1(0,<1,0>)}\alpha_{0,2}^{g_2(0,<1,0>)}} \\ &= \frac{\alpha_{1,1}}{\alpha_{1,1}+\alpha_{0,1}} \end{aligned}$$

我々は素性として，前文節，後文節，2 文節間それぞれの持つ属性およびそれらの組合せを考える．実際に実験で用いた素性については 4 章で述べる．

これまでの多くの先行研究と同様にそれぞれの係り受けは独立であると仮定し☆，一文全体の係り受け確率を，その文中にあるそれぞれの係り受けの確率の積で表す．そして，一文全体の確率が最大となるような

☆ 実際には，たとえば「この薬を，チーズを食べたネズミに与えた」のような場合に二重格のチェックが必要であるなど，一般にはそれぞれの係り受けは独立ではない．しかし，独立性を仮定しないと計算が複雑になり，データスパースネスの問題も生じてくるため，あえて独立性を仮定した．

係り受け関係が正しい係り受け関係であると仮定する．

次の章では，この一文全体の確率が最大となるものを効率良く探索する解析アルゴリズムについて述べる．

3. 解析アルゴリズム

この章では，我々の用いた解析アルゴリズムについて説明する．特徴は文末から文頭に向けての係り受け解析と確率を利用したビームサーチにある．

入力文は形態素解析，文節区切認定まで終わっていると仮定する．解析は次の手順で行う．

手順

(1) 一番最後の文節から係り先を考える．後方係り受けのみを仮定するので，最後の文節は文の主辞になり係り先はない．

(2) 次に1つ前の文節を考える．同じく，後方係り受けのみを仮定するので，最後から2つ目の文節は最後の文節にしか係りえない．

(3) 次に最後から3つ目の文節について考える．この文節の係り先は可能性として，最後から2つ目か，最後の文節かのいずれかである．ME（最大エントロピー法）により学習したモデルより計算される2文節間の係り受け確率をスコアとし，可能性として両方の解析結果をとっておく（**図1**）．

(4) 次に最後から4つ目の文節について考える．解析Aを基にすると，係り受け非交差の原則により，この文節は文節（$N-2$）か最後の文節かの2通りの係り先しか持たない．それぞれの解析のスコアは文節（$N-3$）と文節（$N-2$）の係り受け確率の積で与えられる．一方，解析Bを基にすると係り受け先は3つ考えられる（**図2**）．

(5) このような方法を文頭まで繰り返す．文頭まで解析が終わったら，一番良いスコアの結果を解とする．

解析の途中経過はすべて保持しておくわけではなく，ビームサーチを行う．つまり，ビーム幅を k とするとき途中経過の上位 k 位のみ保持しながら解析する．たとえば，上記の例でビーム幅が4であったとし，手順(4)で計算したスコアがそれぞれ0.9，0.8，0.7，0.6，0.5であったとすると，この5つの候補のうちで一番悪いスコア0.5の結果は以降の解析のために保持しないことにする．この判断は，最終的に正しい解析結果はつねに途中経過としても上位 k 位（k はビーム幅）に入っているはずだという仮定に基づく．

	文節 $(N-2)$	文節 $(N-1)$	文節 (N)	
解析A	N	N	—	スコアA
解析B	$N-1$	N	—	スコアB

（行列の各要素は係り先の文節番号を表す）

図1 文末から3つ目まで
Fig. 1 Analyzing up to the third segment from the end.

	文節 $(N-3)$	文節 $(N-2)$	文節 $(N-1)$	文節 (N)	
解析Aa	N	N	N	—	スコアAa
解析Ab	$N-2$	N	N	—	スコアAb
解析Ba	N	$N-1$	N	—	スコアBa
解析Bb	$N-1$	$N-1$	N	—	スコアBb
解析Bc	$N-2$	$N-1$	N	—	スコアBc

図2 文末から4つ目まで
Fig. 2 Analyzing up to the fourth segment from the end.

4. 実験結果と考察

この章では，係り受け解析の実験をいろいろな角度から分析する．実験に用いたコーパスは，京大コーパス（Version 2）[6]の一般文の部分で，基本的に学習には1月1日と1月3日から8日までの7日分（7,958文），試験には1月9日の1日分（1,246文）を用いた．学習のためのツールとしては文献7)のものを利用した☆．このツールでは学習の繰返しの数を設定する必要があり，以下にあげる実験ではすべてとりあえず400に固定した．

以下の節では，まず係り受け解析の実験に用いた素性と実験の結果を示し，続けて，我々の実験内で得られた興味深いデータを紹介する．そして最後に関連研究との比較を行う．

4.1 実験結果

まず，係り受け解析の実験に用いた素性を**表2**，**表3**に示す．表2にあげた素性は素性名と素性値から成り，一文中の2つの文節に着目したとき，それぞれの文節（前文節と後文節）が持ちうる属性あるいは2文節間に現れうる属性を表している．その各々を基本素性と呼ぶことにする．これらの基本素性は文献8)で使われていたものを基に，一般に係り受け解析に有効であろう素性を追加したものである．一方，表3にあげた素性は基本素性の組合せであり，それぞれ基本素性の番号の組で表している．これらの組合せは，筆者らが係り受け解析に有効だと判断したものである．たとえば，前文節の語形部分，後文節の主辞の品詞，文節間の距離がそれぞれ何であるかによって2つの文節が係

☆ 現在このツールは公開されていない．

表2 学習に利用した素性（基本素性）
Table 2 Features (basic features).

基本素性（43種類）			
素性番号	素性名	素性値	削除したときの精度
1	前文節主辞見出し	（2204個）	86.98% (−0.16%)
2	前文節主辞品詞 (Major)	動詞 形容詞 名詞 助動詞 接続詞 ...（11個）	86.43% (−0.71%)
3	前文節主辞品詞 (Minor)	普通名詞 サ変名詞 数詞 程度副詞 ...（24個）	
4	前文節主辞活用 (Major)	母音動詞 子音動詞カ行 ...（30個）	87.14% (±0%)
5	前文節主辞活用 (Minor)	語幹 基本形 未然形 意志形 命令形 ...（60個）	
6	前文節語形 (String)	こそ こと そして だけ と に も ...（73個）	69.73% (−17.41%)
7	前文節語形 (Major)	助詞 接尾辞 子音動詞カ行 判定詞 ...（43個）	
8	前文節語形 (Minor)	格助詞 基本連用形 動詞接頭辞 ...（102個）	
9	前文節助詞 1 (String)	から まで のみ へ ねえ ...（63個）	87.11% (−0.03%)
10	前文節助詞 1 (Minor)	（無）格助詞 副助詞 接続助詞 終助詞（5個）	
11	前文節助詞 2 (String)	けど まま や よ か ...（63個）	87.08% (−0.06%)
12	前文節助詞 2 (Minor)	格助詞 副助詞 接続助詞 終助詞（4個）	
13	前文節句読点の有無	（無）読点 句点（3個）	85.47% (−1.67%)
14	前文節括弧開の有無	（無）「 ‘ （ “ 『 < 【 《 ...（14個）	87.12% (−0.02%)
15	前文節括弧閉の有無	（無）」 ’ ） ” 』 > 】 》 ...（14個）	87.10% (−0.04%)
16	後文節主辞見出し	素性番号1の素性値と同じ（2204個）	86.31% (−0.83%)
17	後文節主辞品詞 (Major)	素性番号2の素性値と同じ（11個）	76.15% (−10.99%)
18	後文節主辞品詞 (Minor)	素性番号3の素性値と同じ（24個）	
19	後文節主辞活用 (Major)	素性番号4の素性値と同じ（30個）	87.14% (± − 0%)
20	後文節主辞活用 (Minor)	素性番号5の素性値と同じ（60個）	
21	後文節語形 (String)	素性番号6の素性値と同じ（73個）	86.06% (−1.08%)
22	後文節語形 (Major)	素性番号7の素性値と同じ（43個）	
23	後文節語形 (Minor)	素性番号8の素性値と同じ（102個）	
24	後文節助詞 1 (String)	素性番号9の素性値と同じ（63個）	87.16% (+0.02%)
25	後文節助詞 1 (Minor)	素性番号10の素性値と同じ（5個）	
26	後文節助詞 2 (String)	素性番号11の素性値と同じ（63個）	87.11% (−0.03%)
27	後文節助詞 2 (Minor)	素性番号12の素性値と同じ（4個）	
28	後文節句読点の有無	素性番号13の素性値と同じ（3個）	84.62% (−2.52%)
29	後文節括弧開の有無	素性番号14の素性値と同じ（14個）	86.87% (−0.27%)
30	後文節括弧閉の有無	素性番号15の素性値と同じ（14個）	86.85% (−0.29%)
31	文節間距離	A（1）B（2〜5）C（6以上）（3個）	84.64% (−2.50%)
32	文節間読点の有無	無 有（2個）	86.81% (−0.33%)
33	文節間 “は” の有無	無 有（2個）	86.96% (−0.18%)
34	文節間括弧開閉の有無	無 開 閉 開閉（4個）	86.08% (−1.06%)
35	文節間前文節同一語形の有無	無 有（2個）	86.99% (−0.15%)
36	文節間前文節同一語形文節主辞品詞 (Major)	素性番号2の素性値と同じ（11個）	
37	文節間前文節同一語形文節主辞品詞 (Minor)	素性番号3の素性値と同じ（24個）	
38	文節間前文節同一語形文節主辞活用 (Major)	素性番号4の素性値と同じ（30個）	
39	文節間前文節同一語形文節主辞活用 (Minor)	素性番号5の素性値と同じ（60個）	
40	文節間後文節同一主辞の有無	無 有（2個）	86.75% (−0.39%)
41	文節間後文節同一主辞文節の語形 (String)	素性番号6の素性値と同じ（73個）	
42	文節間後文節同一主辞文節の語形 (Major)	素性番号7の素性値と同じ（43個）	
43	文節間後文節同一主辞文節の語形 (Minor)	素性番号8の素性値と同じ（102個）	

り受け関係にあるか否かがある程度決まると考えられる．そこで，このような基本素性の組合せ，たとえば「前文節語形 (String):に&後文節主辞品詞 (Major):動詞&文節間距離:A(1)」を1つの素性として扱うことにした．素性の総数は約60万個である．そのうち学習には学習コーパスで3回以上観測された素性40,893個を用いた．

表2の素性名で使われている用語の意味は以下のとおりである．

主辞 各文節内で，品詞の大分類が特殊，助詞，接尾辞となるもの☆を除き，最も文末に近い形態素．

語形 各文節内で，特殊を除き最も文末に近い形態素．もしそれが助詞，接尾辞以外の形態素で活用型，活用形☆☆を持つものである場合はその活用部分とする☆☆☆．

助詞1・助詞2 各文節内で，一番文末に近い助詞を「助詞1」，その次に文末に近い助詞を「助詞2」とする．

文節間前文節同一語形文節 着目している2文節間に

☆ これらの品詞分類は JUMAN[9] のものに従う．

☆☆ JUMAN の活用型，活用形に従う．

☆☆☆ 語形は基本的に活用部分を指すが，単独の名詞，副詞などからなる文節の場合には語形部分なしとするのではなく主辞と同じであると考える．このようにするのは，一般に前文節の後ろ部分と後文節の前部分が係り受け関係を決めるのに有効であると考えられているからである．もし語形部分なしとすると，名詞と副詞の違いを学習できなくなる可能性があり，それを避けるためこのように定義した．

表3 学習に利用した素性（基本素性の組合せ）
Table 3 Features (combined features).

	基本素性の組合せ（134種類）	削除したときの精度
2素性	(6,16), (7,16), (8,16), (6,17), (7,17), (8,17), (6,18), (7,18), (8,18)	86.99% (−0.15%)
3素性	(6,17,31), (7,17,31), (8,17,31), (6,18,31), (7,18,31), (8,18,31), (6,17,32), (7,17,32), (8,17,32), (6,18,32), (7,18,32), (8,18,32), (6,17,33), (7,17,33), (8,17,33), (6,18,33), (7,18,33), (8,18,33), (6,17,34), (7,17,34), (8,17,34), (6,18,34), (7,18,34), (8,18,34), (6,17,35), (7,17,35), (8,17,35), (6,18,35), (7,18,35), (8,18,35), (6,17,36), (7,17,36), (8,17,36), (6,18,36), (7,18,36), (8,18,36), (6,17,37), (7,17,37), (8,17,37), (6,18,37), (7,18,37), (8,18,37), (6,17,38), (7,17,38), (8,17,38), (6,18,38), (7,18,38), (8,18,38), (6,17,39), (7,17,39), (8,17,39), (6,18,39), (7,18,39), (8,18,39), (6,17,40), (7,17,40), (8,17,40), (6,18,40), (7,18,40), (8,18,40), (6,17,41), (7,17,41), (8,17,41), (6,18,41), (7,18,41), (8,18,41), (6,17,42), (7,17,42), (8,17,42), (6,18,42), (7,18,42), (8,18,42), (6,17,43), (7,17,43), (8,17,43), (6,18,43), (7,18,43), (8,18,43), (29,30,34), (9,11,17), (9,11,18), (10,12,17), (10,12,18)	86.47% (−0.67%)
4素性	(6,17,13,28), (7,17,13,28), (8,17,13,28), (6,18,13,28), (7,18,13,28), (8,18,13,28), (1,6,16,21), (1,7,16,22), (1,8,16,23), (2,6,17,21), (2,7,17,22), (2,8,17,23), (3,6,18,21), (3,7,18,22), (3,8,18,23), (6,17,35,40), (7,17,35,40), (8,17,35,40), (6,18,35,40), (7,18,35,40), (8,18,35,40)	85.65% (−1.49%)
5素性	(1,6,16,21,31), (1,7,16,22,31), (1,8,16,23,31), (2,6,17,21,31), (2,7,17,22,31), (2,8,17,23,31), (3,6,18,21,31), (3,7,18,22,31), (3,8,18,23,31), (2,9,11,17,21), (2,10,12,17,21), (3,9,11,18,21), (3,10,12,18,21), (2,9,11,17,22), (2,10,12,17,22), (3,9,11,18,22), (3,10,12,18,22), (2,9,11,17,23), (2,10,12,17,23), (3,9,11,18,23), (3,10,12,18,23)	86.96% (−0.18%)

表4 解析結果
Table 4 Results of dependency analysis.

	係り受け正解率	文正解率
本手法（$k=1$）	87.14% (9,814/11,263)	40.60% (503/1,239)
本手法（$k=11$）	87.21% (9,822/11,263)	40.60% (503/1,239)
ベースライン	64.14% (7,224/11,263)	6.38% (79/1,239)
人手による規則	72.57% (8,173/11,263)	14.45% (179/1,239)

あり，前文節の語形部分と同じ語形部分を持つ文節．

文節間後文節同一主辞文節 着目している2文節間にあり，後文節の主辞部分と同じ主辞部分を持つ文節．

主辞見出し 主辞の基本型（単語）．素性値として用いる単語は，以下のようにして抽出したものである．まず，学習コーパスの各文に対し，3章で説明した解析アルゴリズムと同様に文末から文頭に向けて順に2文節（前文節と後文節）を取り上げ，そこに共起する単語のペアを抽出する．そのうち3回以上出現したものについて，そのペアを構成するそれぞれの単語を素性値とする．

次に我々の解析結果を**表4**に示す．ここで，係り受けの正解率というのは文末の1文節を除く残りすべての文節に対して，係り先を正しく推定していた文節の割合を求めたものである．また，文正解率というのは文全体の解析が正しいものの割合を意味する．表4の第1行は京大コーパス1月9日の1,246文に対して，コーパスの形態素情報，文節区切情報を入力として，文節間係り受けの解析を決定的に（ビーム幅 $k=1$）行った結果である．SUN Sparc Station 20を用いた実験では，1文あたりの平均解析時間は0.15秒であった．ビーム幅 k を広くして実験をしてみたが，飛躍的な精度の向上は見られなかった．参考として表4の第2行に，ビーム幅 k を20まで変えたときの実験で最高の精度を出したビーム幅11の結果を示す．これは1章にあげた日本語文の係り受けの特徴（4）をある程度裏付ける結果であるといえる．ベースラインとしては各文節がすべて隣に係るとしたときの精度をあげた．さらに参考として，表4の最後の行には人手で規則を作成しそれを用いて解析したときの精度もあげた．このとき用いた規則を**表5**にあげる．ここで用

表5 人手による規則
Table 5 Hand-made rules.

前文節語形の条件	係り先
の（接続助詞）	前文節の次の文節
指示詞	前文節の次の文節
連体詞	前文節の次の文節
の（格助詞），かつ，読点あり	前文節の次の文節
格助詞	動詞を含む最も近い文節
は（副助詞）	動詞を含む最も遠い文節
連体形	名詞を含む最も近い文節
タ形	名詞を含む最も近い文節
連用形	動詞を含む最も近い文節
テ形	動詞を含む最も近い文節
接続詞	文末の文節
名詞性述語接尾辞，かつ，読点あり	動詞を含む最も近い文節
名詞性名詞接尾辞，かつ，読点あり	動詞を含む最も近い文節
名詞性特殊接尾辞，かつ，読点あり	動詞を含む最も近い文節
名詞性述語接尾辞	前文節の次の文節
名詞性名詞接尾辞	前文節の次の文節
名詞性特殊接尾辞	前文節の次の文節
副詞	動詞を含む最も近い文節
その他	前文節の次の文節

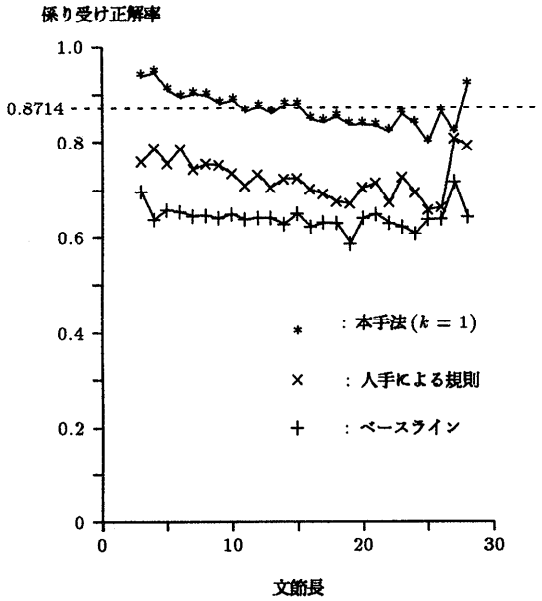


図3 文節長と係り受け正解率（ビーム幅 $k = 1$ のとき）
Fig. 3 Relationship between the number of bunsetsus in a sentence and dependency accuracy (beam breadth $k = 1$).

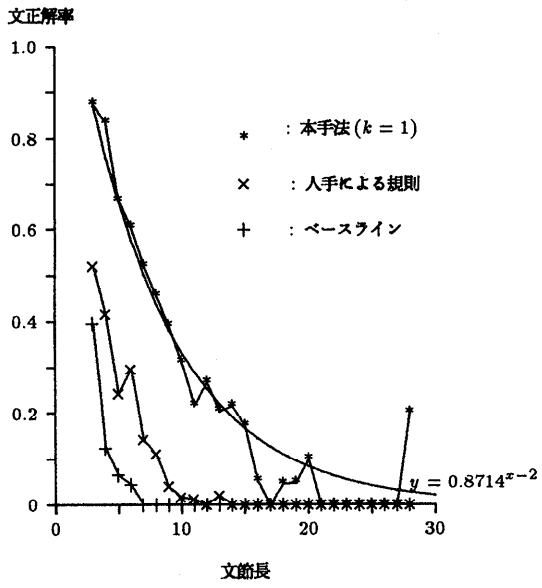


図4 文節長と文正解率
Fig. 4 Relationship between the number of bunsetsus in a sentence and sentence accuracy.

いた方法は，表5の上位から最初に適用可能な規則に従って係り先を決定的に決めるというものである．解析は我々の手法と同様に文末から文頭に向けて行った．

図3，**図4** はそれぞれテスト文の文節長ごとの係り受け正解率，文正解率を求めてグラフ化したものである．**図5** にはテストコーパスにおける文節長と文数の関係を示す．ここで，29文節以上の文はそれぞれ1つしかなかったので結果をプロットしていない．図3からは文節長が長くなっても係り受け正解率の精度は極端に悪くならないことが分かる．一方，図4からは文正解率が $y = 0.8714^{x-2}$ のグラフにほぼ沿っていることが分かる．このグラフは，文節長を x，文正解率を y とするとき，y が係り受け正解率87.14%を $(x-2)$ 個（最後と最後から2番目の文節を除いている）掛け合わせた値で与えられると仮定したときに得られるグラフである．15文節を超えるような文節長の長い文に対しては仮定したグラフに比べて文正解率が悪くなっているが，これは文の先頭付近には副助詞「は」をともなう文節が現れることが多く，長い文ほどその係り先を正しく推定することが難しくなってくるためであると考えられる．今後，「は」をともなう文節の解析に有効と考えられる素性を調査し，追加していきたい．

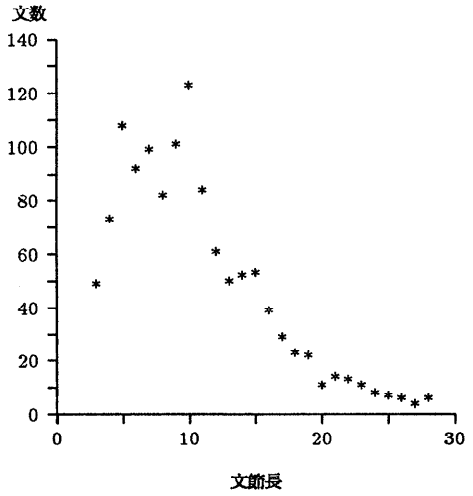


図5 文節長（3文節以上）と文数（合計1,199文）の関係
Fig. 5 Relationship between the number of bunsetsus in a sentence and number of sentences.

一般に従属節の数や並列構造の数が多いと係り受け解析は難しくなる．そこで，**図6** には一文中に含まれる動詞の数と係り受け正解率の関係をグラフ化したものをあげる．テストコーパスにおいて，一文中に含まれる動詞の数の平均値は2.5個であった．このグラフでは厳密に従属節の数を調べたことにはならないが，同様の難しさを表す1つの指標になると考えられる．また，**図7** には一文中に含まれる並列構造の数と係り受け正解率の関係をグラフ化したものをあげる．テストコーパスにおいて，一文中に含まれる並列構造の数

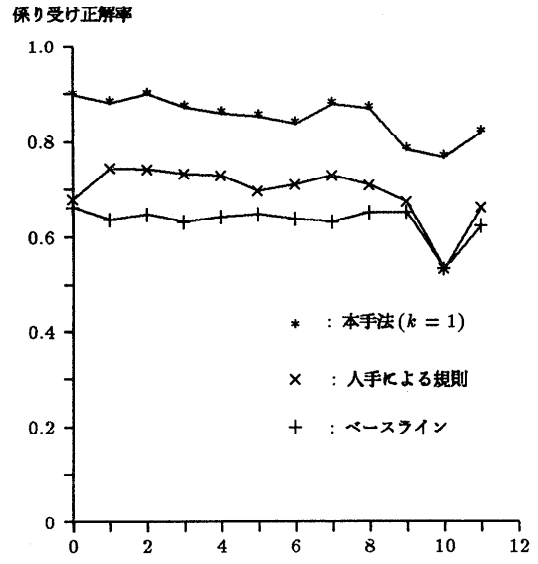


図6 動詞の数と係り受け正解率
Fig. 6 Relationship between the number of verb in a sentence and dependency accuracy.

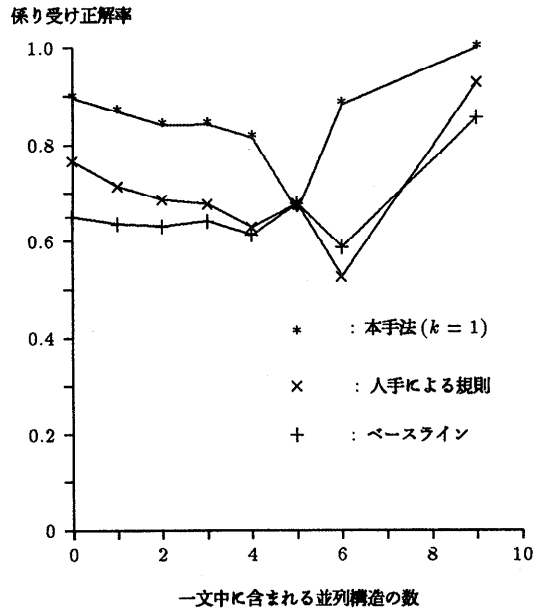


図7 並列構造の数と係り受け正解率
Fig. 7 Relationship between the number of coordinate structures in a sentence and dependency accuracy.

の平均値は 0.69 個であった．これらの図から，一文中に含まれる動詞の数や並列構造の数が多くなっても係り受け正解率の精度は極端に悪くならないことが分かる．

表6 各素性を削除したときの解析精度
Table 6 Accuracy without several types of features.

削除した素性	解析精度
主辞見出し全部	86.30% (−0.84%)
素性 35〜43	86.83% (−0.31%)
素性集合 (1)†	85.71% (−1.43%)
素性集合 (2)‡	86.47% (−0.67%)
4 素性以上の組合せ全部	84.27% (−2.87%)
3 素性以上の組合せ全部	81.28% (−5.86%)
素性の組合せ全部	68.83% (−18.31%)

† (素性 4,5,9〜12,14,15,19,20,24〜27,29,30,34〜43 を削除)
‡ (素性 4,5,9〜12,19,20,24〜27,34〜43 を削除)

試験に用いたコーパスは現在考慮している素性の数に至るまでに何度か試験してその精度だけは確認したが，具体的にどこを間違えてどこを正解したかということは見ていない．素性も一般的なものであり，試験は限りなくオープンに近いものと考えてよい．

4.2 素性と解析精度

この節では，我々が実験で用いた素性のうちいくつかについて，それぞれの素性がどの程度解析精度の向上に貢献しているかを示す．

4.1 節にあげた表 2，表 3 の右欄には，それぞれの素性を削除したときの解析精度と削除したことによる精度の増減を示してある．基本素性を削るときは，それを含む組合せの素性もいっしょに削った．

それぞれの素性だけでなく，複数の素性を一度に削除したときの精度についても調べてみた．これを**表 6** に示す．以上，素性の削除の実験結果はすべて，係り受けを決定的に（ビーム幅 $k = 1$）解析したときのものである．

まず，最も精度に影響していると考えられるのは，前文節の語形と後文節の主辞品詞である．次に，文節間の距離，文節内の句読点の有無，括弧の有無，その次あたりに前文節の主辞品詞，後文節の主辞見出し，後文節の語形が影響していると考えられる．これらの結果は人間の直観にも一致しており，コーパスから人間が書く優先規則のようなものが学習されているといえる．

主辞見出しを考慮することで正解となった例の中には「応じて–決める」，「形で–行われる」などの呼応表現に近いものが多かった．今後，素性としてこのような呼応表現に近い共起単語を選択するようにすればダイレクトに精度向上につながりそうである．

次に組合せの素性を削除したときの精度への影響を見てみると，それぞれの表の結果から，組合せの素性がかなり精度向上に貢献していることが分かる．しかしながら，ME による学習では，自動的に素性間の依

存関係が学習されるわけではない．したがって，MEを利用して素性間の依存関係を学習したいときは，それぞれの素性の組合せを新たな素性として投入する必要がある．しかし，素性を加えることによって精度が下がることもあるので，すべての素性の組合せを新たな素性として投入してするのはあまり得策ではない．それに，基本素性の数が多ければ，組合せの素性の数は爆発的に増えるため，すべての組合せを投入すると現在のマシンパワーでは学習が終わらない．現在我々が考慮している基本素性は数が多いため，上の理由からすべての組合せを投入することはできない．また，直観的に重要そうである基本素性の組合せはある程度推測できるとも判断したため，今回は組合せを人手で選択した．しかし，必ずしも重要な組合せを網羅できているとは限らないため，今後，重要な素性の選択についても検討する必要があると考えている．

素性選択の単純な方法としては学習コーパスにおける素性の出現頻度によって間引く方法が考えられるが，これまでの実験ではこの方法をとるといつも精度が下がった．重要そうな素性を高速に選択する近似解法も提案されている[10]が，精度の低下は避けられないようである．他にも素性の選択に関しては，Berger ら[11]が興味深い提案をしている．

実験で我々の手法が間違った文を調べてみると，間違いの部分に並列構造が絡んでいることが多いということが分かった．そこで，並列構造解析の部分がうまくできるようになった場合，少なくともどの程度の精度向上が期待できそうかということを調べるため，次のような実験を行った．並列句の類似性に着目して並列構造を精度良く検出するルールベースの日本語構文解析器に KNP[12]がある．この KNP の並列構造解析結果を我々の手法に採り入れてみる．KNP は係り先の文節が並列の場合にはその情報（並列ラベル P）も同時に出力する．そこで，KNP で解析した係り受けのうち係り先が並列と出力された文節についてはその解析結果を，完全に答とするのではなく，優先させるようにして実験した．実際には，KNP が並列のラベル P を出力した文節については，我々の解析手法における係り受け確率を 0.9999 とすることによって KNP の解析結果を優先させた．ビーム幅は 10 とした．我々が試験に用いた 1,246 文，学習に用いたコーパスの 1 月 8 日分（1,202 文）の中から，KNP の文節区切認定と一致する 1,161 文，1,100 文をそれぞれ取り出して実験したところ，**表 7** の「Joint」の欄に示される結果を得た．ただし，括弧内は KNP の並列構造の推定がすべて正しかったとしたときの精度である．

表 7 KNP の並列構造解析結果を優先させた場合の精度
Table 7 Accuracy using coordinate structure information given by KNP.

コーパス	係り受け正解率		
	本手法	KNP	Joint
テスト	87.33% (8987/10291)	89.87%	88.40 (89.66)%
学習	90.85% (8180/ 9004)	89.27%	90.83 (92.34)%

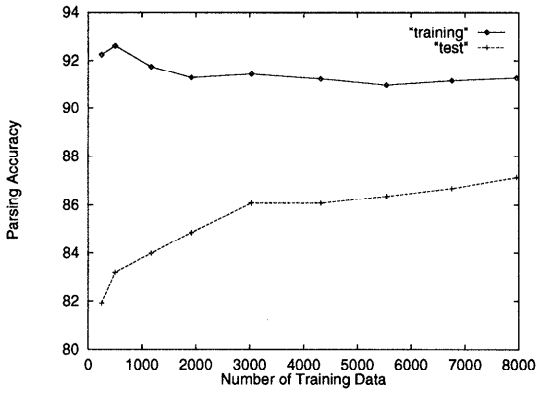


図 8 学習コーパスの量と精度の関係（ビーム幅 $k = 1$）
Fig. 8 Relationship between the number of training data and the parsing accuracy (beam breadth $k = 1$).

表 7 にあげた結果からも分かるように，並列構造を学習できれば少なくとも 2%程度は精度が向上すると期待できる．我々は新たな素性を投入することによって並列構造もある程度学習できるようになると考えている．そのためには，一文全体にわたる広い範囲の情報を獲得するような素性も投入する必要がありそうである．

4.3 学習コーパスと解析精度

この節では，学習コーパスと解析精度の関係について考察する．まず，**図 8** に学習コーパスの量と精度の関係をあげる．この図には学習コーパスとテストコーパスのそれぞれを解析した場合のコーパスの量と解析精度の関係を載せている．学習コーパスに対する実験としては京大コーパス 1 月 1 日の 1,172 文を用いた．

学習コーパスが 1,172 文という少ない量でもテストコーパスに対して 84.0%の精度が出ており，ME がデータスパースネスに強いことが分かる．また，テストコーパスに対する学習曲線から推測すると，学習コーパスの量が増えるともう少し精度が良くなりそうである．

4.4 関連研究との比較

この節では統計的な手法を用いて日本語係り受けの問題に取り組んでいる他の研究との比較を行う．

白井ら[13]との比較

白井らは EDR コーパス，RWC コーパス，京大コー

パスを用いて，構文的な統計情報と語彙的な統計情報をそれぞれ独立なモデルで学習している．語彙モデルでは，たとえば2個あるいは3個の助詞が同じ動詞に係る確率を推定するときにMEを用いる．さらに推定のために有効な素性の取捨選択にもMEを用いている．実験には京大コーパスの文節長7〜9の文からランダムに選んだ500文を対象とし，84.34%の精度を出している．一方，我々の実験では試験コーパスの中で文節長が7〜9の文，303文における結果は87.23%であった．ここでは文末から2つ目の文節は評価から除いており，白井らの方法も文節区切認定が終わった状態からの解析である．対象の文が完全に一致しておらず，対象の文の選択の方法も異なるので，参考にしかできないが，この文節長の文に対しては，我々の手法は白井らの手法に比較して3%程度精度が良かった．また，我々の実験では白井らの実験に比べてそれほど広範囲のデータを学習データとして利用していないが，その割には高い精度が得られている．

江原[14]との比較

江原はMEに基づくモデルを用いて係り受けにある2文節（正例）とない2文節（負例）それぞれの確率分布を用いて2文節間の係り受けの整合度というものを定義している．それぞれの確率モデルはMEに基づいており，素性としては我々と同様に前文節と後文節それぞれが持ちうる属性あるいは2文節間に現れうる属性を用いている．江原の手法と我々の手法との間には，素性の数に大きな差がある．その理由は，江原が2つの組合せまでを用いているのに対し，我々は3つ，4つ，5つの組合せも用いているためである．4.2節にも示したように，2つだけでなく，3つ以上の素性の組合せを用いると5%以上精度が良くなることから，この組合せの素性の違いが我々の手法との精度の違いの原因の1つとなっていると考えられる．しかし，江原の手法で対象としている文はNHKのニュース原稿であり，平均文節長も17.8と我々の対象にしている京大コーパス（平均文節長は10.0）とはまったく異なっている．当然，係り先の候補の数が多い分難しいといえる．また，我々の手法では文末から解析するため，文末から順に各文節を見ていくとき，各段階で非交差条件を満たす係り先の候補がいくつかに絞られるという点で有利である．したがって，単純な比較はできない．

藤尾ら[15]，春野ら[8]との比較

藤尾らは文節間の属性の共起頻度による統計的解析手法を提案した．また，春野らは決定木およびブースティングを利用した係り受け解析を行っている．我々と同様，一文全体の係り受け確率は，一文を構成する個々の文節とその係り先との間の係り受け確率の積から求まると仮定しており，2文節の間の係り受けの確率を計算するための係り受け確率モデルを採用している．これらの評価はEDRコーパスを利用し，試験対象データの選択手法も我々とは異なっているため，直接的な評価は難しい．しかし，ともに85%程度の正解率が出ており，我々の手法とも同様な位置を占めている．

藤尾らや春野らが用いている属性とほぼ同じ属性を用いて我々の手法で実験したところ結果は，それぞれ表6の素性集合（1），素性集合（2）の欄にあげる精度となった．ともに85%以上の精度が出ており，学習コーパスの量に10倍以上の差があることも考えると，MEに基づく確率モデル，文末からの解析手法を用いるとかなり良い精度を出せることが分かる．また，表2，表3の素性をすべて用いたときには，もう少し良い精度が出ていることから，素性をどんどん投入していくことにより精度が向上することも期待できる．

5. まとめ

本論文ではMEに基づくモデルを利用した統計的日本語係り受け解析手法について述べた．一文全体の係り受け確率は，一文中のそれぞれの係り受けの確率の積から求められると仮定し，それぞれの係り受けの確率はMEによって学習した係り受け確率モデルから計算する．この確率モデルは，学習コーパスから得られる情報を基に2つの文節が係り受け関係にあるか否かを予測するのに有効な素性を学習することによって得られる．

我々が素性として利用した情報のうちそれぞれを削除した実験を行うことによって，前文節の語形と後文節の主辞品詞およびそれらの組合せの素性が係り受けの解析には特に重要な情報であること，我々の考慮している素性が精度の向上に貢献していることが分かった．また，学習コーパスの量を変えてみる実験を行うことによって，我々の手法が少ない学習データに対しても有効であることも分かった．係り受けの正解率は，京大コーパスを使用した実験で係り受け正解率が87.1%と高い精度を示している．これは一文全体の係り受けを文末から文頭の方向へ向かって決定的に解析した場合に得られた精度である．このように文末から解析を行うという手法とMEに基づく確率モデルを組み合わせることにより，精度良く日本語の係り受けを解析することができる．

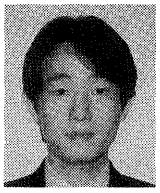
謝辞 MEについて有益な助言をくださったニュー

ヨーク大学の Andrew Borthwick 氏に，心から感謝の意を表する．

参考文献

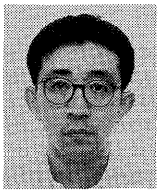
1) Collins, M.: A New Statistical Parser Based on Bigram Lexical Dependencies, *Proc. 34th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp.184–191 (1996).
2) Ratnaparkhi, A.: A Linear Observed Time Statistical Parser Based on Maximum Entropy Models, *Proc. Empirical Method for Natural Language Processings* (1997).
3) 江原暉将：係り受け整合度を計算するいくつかの統計的手法の比較，情報処理学会自然言語処理研究会，Vol.NL126-4, pp.25–30 (1998).
4) 藤田克彦：決定的係り受け解析に関する試み，昭和 63 年度人工知能学会全国大会，pp.399–402 (1988).
5) Pietra, S.D., Pietra, V.D. and Lafferty, J.: Inducing Features of Random Fields, Technical Report, CMU-CS-95-144, Carnegie Mellon University (1995).
6) 黒橋禎夫，長尾 眞：京都大学テキストコーパス・プロジェクト，言語処理学会第 3 回年次大会，pp.115–118 (1997).
7) Ristad, E.S.: Maximum Entropy Modeling Toolkit, Release 1.6 beta (1998). http://www.mnemonic.com/software/memt
8) 春野雅彦，白井 諭，大山芳史：決定木を利用した日本語係り受け解析，自然言語処理シンポジウム '97「実用的な自然言語処理に向けて」(1997). http://www.csl.sony.co.jp/person/nagao/nlsym97/
9) 黒橋禎夫，長尾 眞：日本語形態素解析システム JUMAN 使用説明書 version 3.5，京都大学大学院工学研究科 (1997).
10) 白井清昭，乾健太郎，徳永健伸，田中穂積：最大エントロピー法による確率モデルのパラメタ推定に有効な素性の選択について，言語処理学会第 4 回全国大会，pp.356–359 (1998).
11) Berger, A. and Printz, H.: A Comparison of Criteria for Maximum Entropy/Minimum Divergence Feature Selection, *Proc. 3rd Conference on Emprical Methods in Natural Language Processing*, pp.97–106 (1998).
12) 黒橋禎夫：日本語構文解析システム KNP 使用説明書 version 2.0b6，京都大学大学院情報学研究科 (1998).
13) 白井清昭，乾健太郎，徳永健伸，田中穂積：統計的構文解析における構文的統計情報と語彙的統計情報の統合について，自然言語処理，Vol.5, No.3, pp.85–106 (1998).
14) 江原暉将：最大エントロピー法を用いた日本語文節間係り受け整合度の計算，言語処理学会第 4 回年次大会，pp.382–385 (1998).
15) 藤尾正和，松本裕治：統計的手法を用いた係り受け解析，情報処理学会自然言語処理研究会，Vol.NL117-12, pp.83–90 (1997).

(平成 10 年 11 月 16 日受付)
(平成 11 年 7 月 1 日採録)



内元 清貴（正会員）

1994 年京都大学工学部電気工学第二学科卒業．1996 年同大学院修士課程修了．同年郵政省通信総合研究所入所，郵政技官．自然言語処理の研究に従事．言語処理学会，ACL 各会員．



関根 聡（正会員）

1987 年東京工業大学応用物理学科卒業．同年松下電器東京研究所入社．1990～1992 年 UMIST, CCL, Visiting Researcher. 1992 年 MSc. 1994 年から New York University, Computer Science Department, Assistant Research Scientist. 1998 年 PhD. 同年から Assistant Research Professor. 自然言語処理の研究に従事．人工知能学会，言語処理学会，ACL 各会員．

井佐原 均（正会員）

1978 年京都大学工学部電気工学第二学科卒業．1980 年同大学院修士課程修了．工学博士．同年通商産業省電子技術総合研究所入所．1995 年郵政省通信総合研究所関西支所知的機能研究室室長．自然言語処理，機械翻訳の研究に従事．言語処理学会，人工知能学会，日本認知科学会，ACL 各会員．