統計的部分構文解析器のふるまいについて

乾孝司 *1　木村啓 *1　乾健太郎 *1 *2

*1 九州工業大学情報工学部知能情報工学科

*2 科学技術振興事業団さきがけ研究 21

{k_inui,a_kimu,inui}@pluto.ai.kyutech.ac.jp

1 はじめに

近年，大規模コーパスの利用によって構文解析の精度を向上させる試みが盛んに行われており，報告されている性能も徐々に上がってきている．たとえば，Wall Street Journal による実験では labelled precision が 86% を越えたと報告されており [1]，日本語でも新聞記事を対象とした係り受け解析の実験で 85% ～ 90% の精度が得られたという報告がある [6, 4]．構文解析で十分に高い精度が得られれば，格フレームなどの学習に必要な共起データをプレーンテキストから自動的にかつ大量に抽出することができるようになり，言語解析システムのブートストラップ的な洗練を実現する基礎を与えることができる．また，情報抽出や要約などの応用的タスクにおいても，高精度な構文解析は理解に基づく手法の開発を進める際に不可欠な要素技術である．しかしながら，解析精度が 90% 前後に留まっている現状では，現在の構文（係り受け）解析技術がこれらの需要に十分に応えているとは言いがたい．その一方で，構文解析という作業が実際には意味解釈や省略補完などの作業と不可分であることを考えると，現在盛んに研究されている統計的構文解析技術が解析精度を飛躍的に向上させることも期待しにくい．

このような背景から，我々は統計的部分構文解析方式を提案している [5]．本方式では，文節間の係り受け構造のような依存構造の解析を前提とする．確率言語モデルを用いて個々の部分的な依存関係の確信度を計算し，十分に高い確信度をもつ依存関係だけを選択的に決定することにより部分解析を実現する．個々の依存関係の確信度は，確率言語モデルに基づいて順序づけされた文全体の依存構造の上位 n 個の候補による重みつき多数決によって決まる．

この方式には次のような利点があると期待できる．

まず，解析器を利用する側から見ると，解析器の出力の精度を用途に応じて任意の高さに設定することができるという利点が期待できる．確信度の高い依存関係だけを決定するようにすれば，解消できる曖昧性の数は減るが，高い精度が得られる．逆に，確信度の低い依存関係も決定するようにすれば，精度は落ちるが，決定できる依存関係の数は増える．このようなトレードオフを利用者が選択できるようになれば，現状の不完全な統計的構文解析器でも用途が広がると考えられる．

一方，解析器を洗練・拡張する開発者 (技術者) 側から見ると，解析器の問題点の検出に有用な情報が得られるという利点が期待できる．我々の方式では，従来のように 1 位の解析結果だけを分析する方法とは異なり，上位 n 個の解析結果を横並びで比較する．これによって，1 位の解析結果だけからでは得られない豊富な情報を引き出すことが可能になる．得られた情報は，解析精度の評価，誤りの分析，コーパスのデバッグといった作業に利用できると考えられる．

ただし，これらの議論はいずれも我々の直観に基づく仮説にすぎない．そこで，開発中の解析器を例にとり，統計的部分構文解析のふるまいを分析することによってこれらの仮説の検証を試みた．本稿では，その結果を報告する．

以下，2節では部分構文解析方式の詳細説明をおこなう．3節では部分構文解析器の利用者側からみた有効性を検証する．また4節では解析器の開発者側からみた有効性を検証する．

2 統計的部分係り受け解析

分析と議論を簡単にするため，以下では文節間の係り受け解析に問題をしぼって議論する．ただし，ここで述べる統計的部分解析方式は，一般の依存構造解析に容易に拡張することができる．

次のような文節切りされた品詞タグつき入力文に対し，その係り受け構造を決定するタスクを考える．

(1) [政界にも]$_1$ [二十代、]$_2$ [三十代の]$_3$ [若者が]$_4$ [飛び込み]$_5$ [「戦後政治」の]$_6$ [幕が]$_7$ [上がりました。]$_8$

i 番目の文節を b_i とすると，この文の正解の係り受け構造は，

係り	b_1	b_2	b_3	b_4	b_5	b_6	b_7
受け	b_5	b_3	b_4	b_5	b_8	b_7	b_8

のような係り文節から受け文節への写像関係として表現できる．この入力文に対し，たとえば，我々が開

表 1: 例文 (1) の解析結果

係り		b_1	b_2	b_3	b_4	b_5	b_6	b_7	$P(R_i)$
受け	R_1	b_8	b_8	b_4	b_5	b_8	b_7	b_8	5.11e-31
	R_2	b_5	b_3	b_4	b_5	b_8	b_7	b_8	1.32e-31
	R_2	b_5	b_4	b_4	b_5	b_8	b_7	b_8	1.01e-31
	R_4	b_5	b_5	b_4	b_5	b_8	b_7	b_8	7.36e-32
	R_5	b_2	b_8	b_4	b_5	b_8	b_7	b_8	7.35e-32
	R_6	b_8	b_8	b_4	b_5	b_6	b_7	b_8	9.76e-33
	R_7	b_8	b_8	b_4	b_5	b_7	b_7	b_8	8.95e-33
	R_8	b_2	b_3	b_4	b_5	b_8	b_7	b_8	4.71e-33
	R_9	b_2	b_4	b_4	b_5	b_8	b_7	b_8	3.61e-33
	R_{10}	b_2	b_5	b_4	b_5	b_8	b_7	b_8	3.48e-33

発中の統計的係り受け解析システム [4] は表 1 のような解析結果を出力する．ここで，R_i は終端記号列を入力文とする係り受け構造の i 番目の候補である．係り受け構造の候補は統計的係り受け解析システムが持つ確率言語モデル $P(R_i)$ によってランキングされている[1]．

係り受け解析（構文解析）システムを評価する際には文節ベースの係り先正解率やラベルつき再現率 / 適合率 (labelled recall/precision) など，1 位の候補と正解の重なりの割合を定量化するのが一般的である．文節ベースの係り先正解率によると，上の例では 1 番目の文節と 2 番目の文節の係り先が誤りであり，他の文節の係り先は正しいので，文節ベースの正解率は $4/6 = 67\%$ と計算される．以下，これを部分係り受け解析に対比させて総係り受け解析と呼ぶ，総係り受け解析では 1 位の候補だけが評価の対象となるので，研究者の注意は 1 位の候補に集中しがちになるが，表 1 のように 2 位以下の候補も一緒に横に並べてみると，より多くの情報がそこから引き出せることがわかる．たとえば，b_1 の係り先の候補を見ると，b_2，b_5，b_8 の 3 つが有力で，いわばシステムがその判断に「迷っている」と言うことができる．これに対し，b_3 の係り先については，上位 10 位までの候補がいずれも b_4 で一致しており，システムがその判断に「自信を持っている」ことがわかる．このように，上位 n 位の候補を横並びで見ると，各文節の係り先の候補についてシステムがどの程度確信をもっているかを知ることができる．この「確信度」と呼べるような量をうまく見積ることができれば，確信度の高い係り受け関係だけを選択的に決定し，残りの部分の判断を保留することができるようになる．

[1]実装上また効率性の都合で，すべての候補に共通する周辺分布は $P(R_i)$ の計算に含まれていないので，表 1 の $P(R_i)$ は正確な確率を表しているわけではない．

表 2: 表 1 から計算される係り受け確率と確率最大の係り受け構造

係り	b_1	b_2	b_3	b_4	b_5	b_6	b_7
受け正解	b_5	b_3	b_4	b_5	b_8	b_7	b_8
R^*	b_8	b_8	b_4	b_5	b_8	b_7	b_8
$P(r\|s)$	.57	.65	1.00	1.00	.98	1.00	1.00

このことは以下のように定式化できる．ある確率言語モデルが生成する係り受け構造の集合を $\mathcal{R}$ とし，その確率分布を $P : \mathcal{R} \mapsto [0, 1]$ $\left(\sum_{R\in\mathcal{R}} P(R) = 1\right)$ とする．さらに，ある係り受け構造 $R \in \mathcal{R}$ の文節 b_i が文節 b_j に係ることを式 $R \models r(b_i, b_j)$，R の終端記号列が文 s であることを式 $R \models s$ で表すことにする．このとき，「入力文 s が係り受け関係 $r(b_i, b_j)$ をもつ」という命題に対する確信度を確率 $P(r(b_i, b_j)|s) = P(r|s) = \frac{P(s,r)}{P(s)}$ として定量化することにする（以下，誤解の恐れのない場合，$r(b_i, b_j)$ を r と略記する）．

$R \models s$ を満たす R のうち確率の高い上位 n 個の集合を $\mathcal{R}_H$，$R \models s$ を満たす R のうち残りの確率の低いもの集合 $\mathcal{R}_L$ とすると，$P(r|s)$ は式(1)で近似推定できる [5].

$$P(r|s) \approx \frac{P^{s\wedge r}_{\mathcal{R}_H}}{P^{s}_{\mathcal{R}_H}} \tag{1}$$

ただし，

$$P^{s}_{\mathcal{R}_H} = \sum_{R\in\mathcal{R}_H : R\models s} P(R) \tag{2}$$

$$P^{s\wedge r}_{\mathcal{R}_H} = \sum_{R\in\mathcal{R}_H : R\models s\wedge r} P(R) \tag{3}$$

$$P^{s}_{\mathcal{R}_L} = \sum_{R\in\mathcal{R}_L : R\models s} P(R) \tag{4}$$

このときの近似誤差 ϵ は高々 $\epsilon \leq \frac{P^s_{\mathcal{R}_L}}{P^s_{\mathcal{R}_H} + P^s_{\mathcal{R}_L}}$ で抑えられるので，適当な大きさの n について $P^s_{\mathcal{R}_H} \gg P^s_{\mathcal{R}_L}$ が成り立つ場合には，近似誤差を無視できる大きさに抑えることができる．式(1) によって得られる係り受け関係 r の確率を以下では r の係り受け確率，あるいは簡単に r の確率と呼ぶ．

(1) にしたがって表 1 の例を解析すると，以下のようになる．R^* は各文節について係り受け確率を最大にする係り先を選択することによって得られる係り受け構造である．すなわち，

$$R^* \models r(b_i, b_j) \Leftrightarrow b_j = \arg\max_{b_j} P(r(b_i, b_j)|s) \tag{5}$$

$P(r|s)$ は R^* の個々の係り受け関係の確率である．R^* では，係り先確率の高い文節 b_3，b_4，b_5，b_6 の係

り先はいずれも正解している．一方，文節 b_1 と b_2 の係り先が誤っているが，これらの係り受け確率はいずれも低いので，実際にはこれらの文節の係り先は判断が保留される．

3 解析器の利用者側から見た利点

実験は，京大コーパス [7] から無作為に抽出した 2,259 文（16,379 文節[2]）をテストデータとしてオープンテストを行った．パーザへの入力は品詞タグつきの単語列である．係り受け確率の推定には，いずれの実験でも統計的構文解析器が出力する上位 100 位の解析木を用いた．

3.1 C-A 曲線

係り受け確率の閾値を σ として，σ 以上の確率をもつ係り受け関係だけを選択的に決定する作業を考える [3]．σ を $.5 \leq \sigma \leq 1$ の範囲で変化させると，図 1 のような被覆率 - 正解率曲線が得られた．以下，これを C-A 曲線と呼ぶ．ただし，被覆率，正解率は次式で与えられる．

$$\text{被覆率} = \frac{\text{係り先が決定された文節の数}}{\text{テストセット中の文節の数}} \quad (6)$$

$$\text{正解率} = \frac{\text{係り先が正解である文節の数}}{\text{係り先が決定された文節の数}} \quad (7)$$

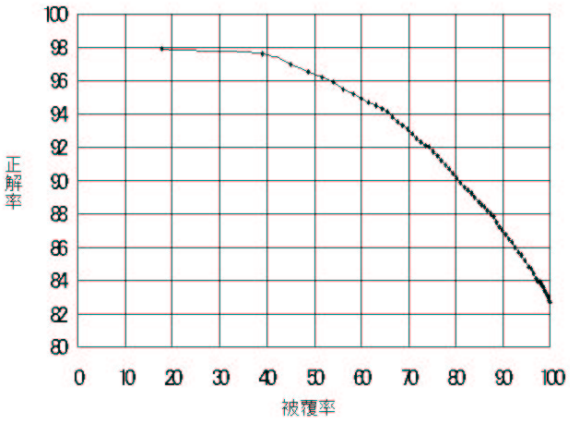

図 1: C-A 曲線

[2]文末の 2 文節は係り先が必ず一意に特定されるので評価の対象には含めない．

[3]厳密に言えば，複数の文節の係り受け関係を同時に選択する場合はそれらの係り受け関係の結合確率を計算するべきである．しかしながら，実際には確率が 1 に近い係り受け関係しか選択されないので，同時に選択された 2 つの係り受け関係に強い負の従属関係があるということは起こりえない．すなわち，確率が 1 に近い係り受け関係だけを選択する限り，その判断の基準としては個々の係り受け関係の確率を参照だけで十分であると考えられる．

図 1 は，確率の高い係り受け関係だけを選択すると文節ベースの正解率が上がることを示している．たとえば，総係り受け解析の正解率は 82.2% に過ぎないが，被覆率を 75% に抑えるだけで正解率は 91.7% に上がり，さらに被覆率を 50% にまで引き下げると 96.2% の正解率が得られる．被覆率はある程度犠牲にしてでも高い精度を必要とする共起データ抽出作業のような場合に統計的部分解析が有効に働くことがわかる．従来の総係り受け解析とは異なり，解析器の用途に応じて被覆率と正解率のどちらを重視するかをユーザが自由に選択できる点が重要である．また，$\sigma = 1$ のときの被覆率が 13% に過ぎない点にも注意したい．このことは，構文的制約で一意に決まる係り受け関係だけを選択的に出力する規則ベースの部分係り受け解析では大部分の文節の係り先が特定されないことを意味している．これに対し，統計的部分解析では，$\sigma = .99$，$\sigma = .98$ などの点を見てもわかるように，わずかなリスクを負うことによって係り受け関係の決定の保留を大幅に減らすことができる．

3.2 係り受け関係の種類と被覆率

図 1 の C-A 曲線にはテストコーパス中の全ての係り受け関係が含まれている．しかしながら，利用者側から見ると，利用者が求めている種類の係り受け関係がこの C-A 曲線のどのあたりに分布しているかが重要な関心事になる．たとえば，要約生成や情報抽出，推敲支援といったアプリケーションでは，並列節や連体修飾節の範囲に関する情報が有用なため，並列節や連体修飾節の範囲推定に寄与する係り受け関係が特定することが重要になる．しかし，言語モデルがこれら重要な係り受け関係に低い係り受け確率しか割り当てられなければ，部分解析を行っても利用者はその利点を享受できない．

そこで，例として，次のような並列節・連体修飾節に係る可能性のある文節を対象として，部分解析器のふるまいを調査した．

文節グループ A： 文末の 2 文節を除く全ての文節．

文節グループ P： 並列節を含む文中で，並列節の述語よりも前方にある文節．並列節の範囲の特定に寄与する．

文節グループ R： 連体修飾節を含む文中で，連体修飾節の述語よりも前方にある文節．連体修飾節の範囲の特定に寄与する．

図 2は，各文節グループごとの最適解の係り受け確率の分布を示している．たとえば，$Pb = 100\text{-}96\%$ は，最適解（すなわち係り受け確率が最大の係り先）の係り受け確率が .96 ～ 1.00 であった文節の割合を表す．一般に，最適解の確率が高い文節が多い方が，

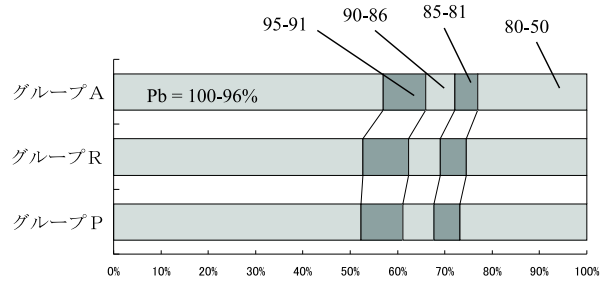

図 2: 最適解の係り受け確率の分布

少ないリスクでより高い被覆率が得られる．図2を見ると，グループ P もグループ R もグループ A に比べるとパフォーマンスが少し低いことがわかる．すなわち，同じ精度を得たい場合に，グループ P やグループ R の文節については係り先決定の被覆率が全体平均よりも悪くなる．ただし，その差はそれほど顕著ではない．他の種類の係り受け関係についても同様に調査したが，特定の種類の係り受け関係だけが被覆率が著しく悪いという現象はとくに見られなかった．このことは，どのような種類の係り受け関係を必要とするアプリケーションであっても，統計的部分解析がある程度有効に機能することを示唆している．

4 解析器の開発者側から見た利点

我々の方式では，従来のように 1 位の解析結果だけを分析する方法とは異なり，上位 n 個の解析結果を横並びで比較する．これによって，1 位の解析結果だけからでは得られない豊富な情報を引き出すことが可能になると考えられる．得られた情報は以下のような作業に有用であると期待できる．

1. 確率言語モデルの評価
2. 解析誤りの分類
3. コーパスの誤り検出

4.1 確率言語モデルの評価尺度

前節で述べた C-A 曲線は，複数の確率言語モデルを比較評価する尺度としても利用することができる．従来の総係り受け解析では被覆率を 100% に固定し，正解率のみでモデルの性能を評価していたが，部分的構文解析では被覆率と正解率の組み合わせによってモデルを評価することになる．被覆率が同じなら正解率が高い方が望ましいし，その逆も同様である．

図 3 は，解析器を改良する過程で作成した 2 種類の確率言語モデルから得られた C-A 曲線である．総係り受け解析（被覆率 100%）の正解率だけを見ると，これらのモデルはともに 82.2% で，性能の違いが見られない．しかしながら，図 3 の C-A 曲線を見ると，二つのモデルの差は明らかである．たとえば，被覆率を 50% に抑えた場合，両者の正解率はそれぞれ 96.2%，95.2% であり，モデル A はモデル B に比べ，誤り率を 20% 削減している．

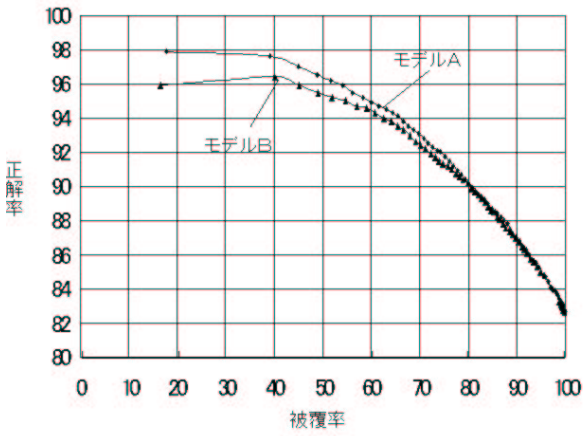

図 3: C-A 曲線 (正解率の差分)

このように，C-A 曲線をプロットすることによって，従来の総係り受け解析による評価では見出せなかったモデルの性質を調べることができる．

4.2 解析誤りの分類

解析器，あるいは言語モデルを洗練するためには，手作業による解析誤り箇所の詳細な分析が不可欠である．しかしながら，実験の規模が大きくなると，解析誤りの数も膨大になり，すべての誤りを網羅的に分析するには多大な人的コストが必要になる．この問題に対しては，分析対象を何らかの基準に基づいてうまく選択し，より重要性の高い誤り箇所を優先して分析するという対策が考えられる．

統計的部分係り受け解析では，各文節について，正解の係り受け確率 (Pc) と最適解の係り受け確率 (Pb) を推定する．これらの情報は，以下に述べるように，誤り箇所の重要性を評価する材料になると考えられる．

最適解が正解と異なる文節を集め，Pb と Pc の組をプロットすると，図 4 のようなグラフが得られた．このなかで，Pb が極端に高く（Pb $\simeq$ 100%），Pc が極端に低い（Pc $\simeq$ 0%）誤りは特に深刻な誤りであると言える（図 4 のグループ L）．Pb が極端に高いことは，部分解析器が最適解に「強い自信をもっている」にもかかわらず，それが正解でなかったことを意味しており，被覆率を犠牲にしてでも高い精度を求める利用者にとって致命的な誤りになる可能性がある．また，Pc が極端に低いことは，解析器が「強い自身をもって」正解を棄却したことを意味しており，言語モデルに深刻な欠陥がある可能性を示唆している．このような理由から，解析器の開発者は，グルー

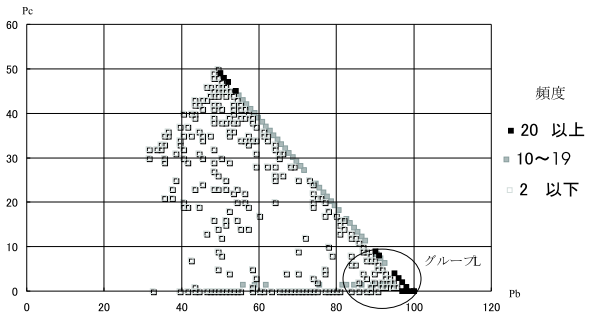

図 4: 解析誤りの分布

プ L の誤りを優先して分析し，解析器の改良を行う必要があると考えられる．

解析器の性能が十分に高いか，解析対象の難易度が低い場合は，グループ L の誤りはそれほど多く生じないと予想できる．その場合は，$Pc \simeq 0$ を固定して，Pb の高い誤り文節から低い誤り文節へと分析の対象を広げればよい．ただし，今回行った解析実験では，実際にグループ L に属す誤りが数多く見られた．たとえば，Pb = 100% かつ Pc = 0% の誤りだけでも 174 箇所（全誤りの約 6%）あり，Pb ≥ 98% かつ Pc = 0% という条件では 580 箇所の誤りが見つかった．

グループ L の誤りの例を以下に示す．誤り分析に当たっては，図 5のような情報を表示する誤り分析ツールを作成し，それを利用した．このツールは，統計的構文解析器 [8] の出力を受け取り，解析精度のサマリを表示するとともに，ユーザが指定する条件に適合する文を検索し，次のような情報を表示する．

- 文節列と文節ラベル列
- 各文節についての正解とその係り受け確率
- 各文節についての最適解とその係り受け確率
- 言語モデルが出力する上位 n 位の係り受け構造とその生成確率
- 各係り受け構造における誤り文節（* で標示）
- 言語モデルに含まれている単語共起の統計量（語彙モデル情報）

文の検索条件には，係り受け関係の種類や Pb，Pc の範囲などを指定することができる．検索された文の集合は，さらに係り受け関係の種類や Pb，Pc の値などでソートすることもできる．

図中の記号 “@” は，ユーザが指定した条件に適合する文節を示している．この例では，文節 (4)「レース数は」に焦点が当たっている．この文節では，Pb = 99%，Pc = 0% で解析誤りが生じている．分析ツールが提供する情報を参照すると，この誤りに関しては以下のように原因を分析することができる．

文節 (4) の正解の係り先は文節 (6) であるが，解析器は Pb = 99% で文節 (9) に係ると判断している．Pc = 0% となった原因は，この例では，文節 (6) の文節ラベルを見れば推測できる．文節 (6) は，判定詞が省略された文節で述語的にふるまうと考えるのが自然だが，現在の言語モデルではこの文節の受け属性が「体言」になっている[4]ため，「レース数は」がそこに係るという解釈は極端に弱くなってしまう．係助詞「は」を伴う文節は，ほとんどの場合に「用言」を受け属性に持つ文節に係るからである．

上述の例では誤り原因を特定することができた．しかし誤り原因にはさまざま要因が影響しあっている場合も考えられ，誤り原因の特定が困難なものもある．図 6は誤り原因の特定が困難な文節の例である．この場合, 文節 (8) の係り先として，文節 (9) の候補が存在しない原因がわからない．

グループ L の誤りから順に Pb 値を下げ，Pb ≃ 50% までの誤りを分析をしていくと，Pb ≃ 50% に近づくほど，誤り原因を特定する作業は徐々に難しくなる．Pb ≃ 50% かつ Pc ≃ 0% の誤りは，部分解析器が最適解として選択した候補に対し「悩んでいる，自信はない」と言っており，さらに悩んでいるもう片方の文節が正答ではないことを意味している．Pb ≃ 100% の誤りでは，分析対象としている誤り文節，正解の文節，誤って係っている文節の 3 文節の属性を比較，検討することで，誤り原因の大部分を特定することができるが，Pb ≃ 50% になると，さらに Pb と同程度の係り受け確率をもつ文節が存在する可能性あり，検討すべき文節は広範囲に広がってしまい，誤り原因の特定は困難なものになる．図 7にその誤り例を示す．この場合，文節 (2) の正解の係り先は文節 (3) であるが，文節 (5) と文節 (10) のあいだで係り先候補が揺れている．表示情報からは文節 (2) の誤り原因を特定することが困難である．

本節では，統計的部分構文解析によって推定した正解の係り受け確率 (Pc) と最適解の係り受け確率 (Pb) の組が，誤り箇所の重要性を評価する材料になるという仮説を立て，実際の分析例を交えながら，部分解析の性質について述べた．ただし，分析の規模が不十分であるため，今後はさらに分析規模を大きくしていく必要がある．

[4]文節ラベル「体_用体_t1」は，受け属性が「体言」，係り属性が「用言」あるいは「体言」であり，末尾に読点が付いていることを表す．

```
$950101149-010 (209) [99] <0>
comment# 京大コーパス id (ツールが持つ文 id)[統合モデル Pb]<統合モデル Pc>

表記    : 1) この間、 2) 挑戦者チームが 3) 戦わなければならない 4) レース数は 5) 最多で 6) ４ ９、
表記    : 7) 約４カ月 8) かけて 9) 争う。
ラベル  : 1) 体 _ 用体 _t1 2) 体 _ 体格 _t0:が 3) 用格 _ 体 _JYvit0 4) 体 _ 体格 _ は t0:は 5) 体 _ 体格 _t0:で
ラベル  : 6) 体 _ 用体 _t1 7) 体 _ 用体 _t0 8) 用格 _ 用体 _HEvtt0 9) 用格 _ 用体 _SYvtt0

係り文節            :  1   2   3   4@  5   6   7   8
正解                :  3   3   4   6   6   9   8   9
================================================================
統合モデル 最適解  :  9*  9*  4   9*  9*  9   9*
統合モデル Pb       : 84  72  99  99  89  97  78
統合モデル Pc       : 15  27  99   0   9  97  21
=========================================================
構文モデル 1位      :  3   3   4   9*  9*  9   9*  9   P = 1.713822e-13
構文モデル 2位      :  9*  3   4   9*  9*  9   9*  9   P = 1.224159e-13
構文モデル 最適解  :  3   3   4   9*  9*  9   9*
構文モデル Pb       : 33  94  99  97  54  59  71
=========================================================
統合モデル 1位      :  9*  9*  4   9*  9*  9   9*  9   1.29e-51
統合モデル 2位      :  9*  9*  4   9*  9*  9   8   9   3.23e-52
統合モデル 3位      :  3   3   4   9*  9*  9   9*  9   2.06e-52
統合モデル 4位      :  9*  3   4   9*  9*  9   9*  9   1.47e-52
統合モデル 5位      :  3   3   4   9*  6   9   9*  9   6.01e-53

語彙モデル情報 (従属係数)
1位の木 -- D(この間 (****)| ) = 1
1位の木 -- D(数 (2507)| は:争う (23530)) = 0.610
1位の木 -- D(が:で | 争う (235):が:を + φ) = 2.24
1位の木 -- D(４ ９ (****)|) = 1
1位の木 -- D(カ月 (2671)| 用:0) = 5.80
1位の木 -- D(チーム (2422)| が:争う) = 1.22
1位の木 -- D(が |Pc) = 0.195
1位の木 -- D(で |Pc) = 0.0780
1位の木 -- D(チーム (0372)| が:争う) = 6.35
1位の木 -- D(カ月 (2670)| 用:0) = 7.14
1位の木 -- D(数 (2422)| は:争う (23520)) = 0.730
1位の木 -- D(カ月 (2422)| 用:0) = 0.947
1位の木 -- D(最多 (2422)| で:争う) = 1.39
3位の木 -- D(で | 争う (235):が:を + φ) = 0.546
3位の木 -- D(チーム (2422)| が:なる) = 0.856
3位の木 -- D(チーム (0372)| が:なる) = 0.837
3位の木 -- D(が | なる + φ (なる)) = 0.398
5位の木 -- D(で | と) = 0.883
5位の木 -- D(で |Pr) = 0.0349
6位の木 -- D(が | 争う (235):が:を + φ) = 0.654
```

図 5: 分析ツールの表示例 (グループ L)

```
$950105230-004 (957) [100] <0>
表記        : 1) 星野さんが 2) 下山した 3) 三日 4) 昼過ぎまでは、 5) 天気も 6) よく、
表記        : 7) 登山と 8) しては 9) 最高だったと 10) 言う。
ラベル      : 1) 体 _ 体格 _t0:が 2) 用格 _ 用体 _vit0 3) 体 _ 用体 _t0 4) 体 _ 体格 _t1:は
ラベル      : 5) 体 _ 体格 _t0:も 6) 用格 _ 用体 _HEajt1 7) 体 _ 体格 _t0:と 8) 用格 _ 用体 _vtt0
ラベル      : 9) 用格 _ 用体 _INnmt0 10) 用格 _ 用体 _SYvtt0
係り文節                  :  1   2   3   4   5   6   7   8@  9
正解                      :  2   3   4   5   6   9   8   9  10
============================================================
統語モデル 最適解        :  2   3   4   6*  6  10* 10* 10*
Pb                         : 80  58  99  51  99  97  69 100
Pc                         : 80  58  99   0  99   0  30   0
============================================================
統語モデル 1位            :  2   3   4   6*  6  10* 10* 10* 10   1.10e-38
統語モデル 2位            :  2   4*  4   6*  6  10* 10* 10* 10   7.81e-39
============================================================
```

図 6: 分析ツールの表示例 (グループ L) : 誤り原因の特定が困難な文節

4.3 コーパスの誤り検出

確率言語モデルの評価の際に，コーパスの持っている係り受け関係を正答として利用した場合を考える．あるモデルにおいて，誤りが生じにくい文節がもしグループLの誤りを起こしているならば，それはモデルの致命的な誤りであるという可能性の他にコーパスの構文タグ自体が誤っている可能性も考えられる．

そこで，Pcを0%から50%まで，10%ごとに分割し，それぞれの誤り文節各30サンプルを分析することでバグ発生率の調査をおこなった．その結果が表3である．

表 3: バグ発生率の調査結果

Pc	発見したバグの数	文脈依存	不適切
41～50	1/30		
31～40	4/30	1/30	
21～30	1/30	1/30	
11～20	0/30	1/30	1/30
1～10	5/30	1/30	
0	3/30		

$Pc = 0\%$～10% の範囲で8/60個のバグを発見できた．ただし，分析規模が全体で180文節と小さいために，この結果からは部分構文解析がコーパスのデバッグに有効であるとはいえず，さらに大規模なデータによる検討が必要である．

表3の”文脈依存”とは，文外の文脈を考慮しなければ一文のみの情報からは正しい係り先を決定できないもので，正解の候補が人間の直感で二通り考えられるものである．また，”不適切”とは，人間が読んでも不自然さが感じられる文である．解析器の入力には適していない文である．

図8，図9および図10にそれぞれの例文を示す．

5 おわりに

本稿では，統計的部分係り受け解析のふるまいについて 1) 解析器の利用者からの視点，2) 解析器の開発者からの視点，の二通りの視点からの仮説を設け，開発中の解析器を例にとり，そのふるまいの検証を試みた．

その結果，解析器の利用者は部分係り受け解析をおこなうことで，解析器の出力の精度を用途に応じて任意の高さに設定することができることが確認できた．また，解析器の開発者の視点から見た場合，部分係り受け解析によって推定された最適解の係り受け確率(Pb)と正解の係り受け確率(Pc)が解析誤りの分類に有効であると考え，それに基づき誤り分析をおこない，その一例を示した．今回おこなった分析はいずれも小規模なものであるため，今後は，さらに大規模な分析をおこなう必要がある．また，分析に費す人的コストを抑えるためにも，誤り文節分析支援ツールの改良も合わせて検討していく予定である．

謝辞

統計的部分構文解析については，北陸先端大の奥村学氏から示唆に富む助言をいただきました．同氏に感謝いたします．実験に当たっては，東京工業大学で開発された統計的構文解析器を利用させていただきました．同大学の田中穂積氏，白井清昭氏，植木正裕氏，橋本泰一氏に感謝いたします．

参 考 文 献

[1] E. Charniak. Statistical parsing with a context-free grammar and word statistics. In *Proceedings of the 15th National Conference on Artificial Intelligence*, 1997.

[2] K. Inui, V. Sornlartlamvanich, H. Tanaka, and T. Tokunaga. A new formalization of probabilistic LR parsing. In *Proceedings of the 5th International Workshop on Parsing Technologies*, pp. 123–134, 1997. Available from `http://www.cs.titech.ac.jp/tr.html`.

[3] K Jensen, G. E. Heidorn, and S. D. Richardson, editors. *NATURAL LANGUAGE PROCESSING: The PLNLP Approach*. @KAP, 1993.

[4] K. Shirai, K. Inui, H. Tanaka, and T. Tokunaga. An empirical study on statistical disambiguation of japanese dependency structures using a lexically sensitive language model. In *Proceedings of Natural Language Pacific-Rim Symposium*, pp. 215–220, 1997.

[5] 乾健太郎, 白井清昭, 田中穂積, 徳永健伸. 統計基づく部分係り受け解析. 言語処理学会年次大会予稿集, pp. 386–389, 1998.

[6] 黒橋禎夫, 長尾眞. 並列構造の検出に基づく長い日本語文の構文解析. 自然言語処理, Vol. 1, No. 1, pp. 35–57, 1994.

[7] 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. 人工知能学会全国大会予稿集, pp. 58–61, 1997.

[8] 白井清昭. 統計情報を利用した統合的自然言語解析. 博士論文, 東京工業大学, 1998. `http://www.cs.titech.ac.jp/tr.html`.

```
$950106221-012 (1197) [51] <0>
表記       : 1) 「本の 2) 売れ行きは 3) それほどでもなかったのですが、 4) 『よく 5) 書いてくれた』と 6) いう
表記       : 7) 共感の 8) 手紙の 9) 多さに 10) 驚いています」と 11) 豊田さん。
ラベル     : 1) 用_体_t0 2) 体_体格_はt0:は 3) 用格_用体_HEnmt1 4) 用格_用体_ajt0 5) 用格_体格_INvtt0
ラベル     : 6) 用格_体_JYvtt07) 体_体格_のt0:の 8) 体_体格_のt0:の 9) 体_体格_t0:に
ラベル     : 10) 用格_体格_INvtt0 11) 用格_用体_SYnmt0
係り文節                  :  1   2@  3   4   5   6   7   8   9  10
正解                      :  2   3  10   5   6   8   8   9  10  11
================================================================
統合モデル 最適解         :  2  10*  5*  5   6   7*  8   9  10
Pb                         : 78  51  47 100  99  51  90 100 100
Pc                         : 78   0  13 100  99  26  90 100 100
================================================================
統合モデル 1位            :  2  10*  6*  5   6   7*  8   9  10  11   1.70e-42
統合モデル 2位            :  2   5*  5*  5   6   7*  8   9  10  11   1.66e-42
================================================================
```

図 7: 分析ツールの表示例 ($Pb \simeq 50\%$ かつ $Pc \simeq 0\%$ の誤り)

```
$950110037-004 (2124) [46] <2>
表記       : 1) 褒める 2) 先生と、 3) 褒め 4) 上手の 5) 先生は 6) 違います。
ラベル     : 1) 用格_体_JYvtt0 2) 体_体格_t1:と 3) 用格_用体_HEvtt0 4) 用格_体_nmt0
ラベル     : 5) 体_体格_はt0:は 6) 用格_用体_SYvit0
係り文節                  :  1   2@  3   4   5
正解                      :  2   3   4   5   6
============================================
comment # 文節(2)は文節(5)に係るのが正解.
```

図 8: 分析ツールの表示例 (京大コーパスのバグ)

```
$950104065-007 (509) [46] <27>
表記       : 1) その 2) ころ、 3) 同じ 4) コンテストで 5) デビューした 6) タレントの
表記       : 7) 武田真治さんが 8) 細い 9) 体に 10) 女性服を 11) 着て 12) 深夜 13) テレビ番組に 14) 出演、
表記       : 15) 「普通の 16) 男の子」に 17) 火を 18) つけた。
ラベル     : 1) 用_体_t0 2) 体_用体_t1 3) 用格_用体_nmt0 4) 体_体格_t0:で 5) 用格_体_JYvtt0
ラベル     : 6) 体_体格_のt0:の 7) 体_体格_t0:が 8) 用格_用体_ajt0 9) 体_体格_t0:に 10) 体_体格_t0:を
ラベル     : 11) 用格_用体_HEvtt0 12) 体_用体_t0 13) 体_体格_t0:に 14) 用体格_用体_t1 15) 用格_体_nmt0
ラベル     : 16) 体_体格_t0:に 17) 体_体格_t0:を 18) 用格_用体_SYvtt0
係り文節                  :  1   2@  3   4   5   6   7   8   9  10  11  12  13  14  15  16  17
正解                      :  2   5   4   5   6   7  14   9  11  11  14  14  14  18  16  18  18
=========================================================================================
comment # 文節(2)の係り先は文節(5), 文節(14), 文節(18)のどれでも可.
```

図 9: 分析ツールの表示例 (文脈依存)

```
$950103080-011 (374) [43] <16>
表記       : 1) 東は 2) ３４分、 3) ＣＫから 4) ＧＫが 5) パンチした 6) こぼれ球を
表記       : 7) けり込み、 8) 最高の 9) タイミングで 10) 決勝点を 11) 奪った。
ラベル     : 1) 体_体格_はt0:は 2) 体_用体_t1 3) 体_体格_t0:から 4) 体_体格_t0:が
ラベル     : 5) 用格_体_JYvtt0 6) 体_用_JYt0 7) 用格_用体_HEvtt1 8) 用格_体_nmt0 9) 体_体格_t0:で
ラベル     : 10) 体_体格_t0:を 11) 用格_用体_SYvtt0
係り文節                  :  1   2   3@  4   5   6   7   8   9  10
正解                      :  7   7   7   5   6   7  11   9  11  11
================================================================
```

図 10: 分析ツールの表示例 (不適切)