自 然 言 語 処 理 140−20
（2000. 11. 22）

形態素係り受けモデルによる構文解析

森 信介 西村 雅史 伊東 伸泰 荻野 紫穂 渡辺 日出雄
日本IBM 東京基礎研究所
〒242-8502 神奈川県大和市下鶴間 1623-14
mori@trl.ibm.co.jp

あらまし
本論文では、形態素単位の係り受けに基づく言語モデルを提案する。このモデルは、文を係り受け関係にある形態素の列とみなし、各形態素を文頭から順に予測する。ある時点での履歴は部分的な構文解析の結果である。これは、形態素をノードとする木の列であり、提案モデルでは、まず、履歴である木の列のうち次の形態素に係る木の数を予測し、続いて係る木の列から次の形態素を予測する。このモデルを用いた構文解析器を作成し、約1,000文の解析済みコーパスからパラメータを推定し、約100文に対して構文解析実験を行なった結果、係り受け単位で89.9%の解析精度を得た。

キーワード 確率的手法 コーパス 構文解析 確率的言語モデル 音声認識

A Stochastic Parser Based on a Structural Word Dependency Model

Shinsuke MORI, Masafumi NISHIMURA, Nobuyasu ITOH, Shiho OGINO, Hideo WATANABE
IBM Research, Tokyo Research Laboratory, IBM Japan, Ltd.
1623-14 Shimotsuruma Yamatoshi Kanagawaken 242-8502 Japan
mori@trl.ibm.co.jp

Abstract
In this paper, we present a stochastic language model using dependency. This model considers a sentence as a word sequence and predicts each word from left to right. The history at each step of prediction is a sequence of partial parse trees covering the preceding words. First our model predicts the partial parse trees which have a dependency relation with the next word and then predicts the next word from those trees. We prepared about 1,000 syntactically annotated sentences and estimated the parameters of our model. We built a parser based on our model and tested it on about 100 sentences. The accuracy of the dependency relation was 89.9%.

Key Words Stochastic Approach, Corpus, Parsing, Stochastic Language Model, Speech Recognition

1 はじめに

音声認識を端緒とする確率的手法は、自然言語処理の優れた方法論の一つである。実際、音声認識の言語モデルの多くは n-gram モデルであり、英語などの形態素解析器の多くは品詞 n-gram モデルやその拡張に基づいている [1, 2, 3, 4]。形態素解析は自然言語処理の最初の段階であり、確率的形態素解析器の精度は多くの応用に対して十分である。次の段階は文の構造を明らかにする構文解析である。最近、多くの確率的構文解析器が提案され、高い精度が報告されている。しかしながら、現状の精度は多くの応用には十分ではなく、さらなる精度向上が望まれる。

構文解析器の主な応用の一つとして、音声言語理解をめざす音声認識の認識結果の構文解析があろう。構文解析器と音声認識器を組み合わせることを考えた場合、構文解析器が生成的な確率的言語モデルを基礎とすることが望ましい。ここで、生成的とは、すべての可能な文字列に対する生成確率の和が 1 以下であることを意味する。言語モデルが生成的であれば、構文解析器と音声認識器を継目なく組み合わせることが可能になる。つまり、音声認識器の言語モデルを、構造を記述する言語モデルとし、従来の n-gram モデルよりも豊富な情報を用いて認識を行ない、同時に構文解析の結果を出力するのである。このような組合せが現実的でない場合でも、音声認識器が N-best の認識結果をその確率とともに出力し、構文解析器がそれらをすべて構文解析するとともに確率値を更新し、この結果得られる確率最大の文とその構造の組を出力することで、ほぼ同じ効果が得られる。したがって、音声認識などの他の確率的手法による言語処理との親和性を考慮した場合、確率的構文解析器の言語モデルは生成的であることが望ましい。

本論文では、日本語を対象言語とする生成的な確率的言語モデルとそれに基づく構文解析器を提案する。このモデルでは、文は形態素の列とみなされ、それぞれの形態素は文頭から順に予測される。予測の各段階での履歴は、その時点までの形態素列を覆う部分解析木の列である。我々のモデルでは、まず、次の形態素に係る部分解析木を予測し、それから、次の形態素をそれに係る部分解析木から予測する。日本語では、それぞれの形態素は必ずその後に出現する形態素に係るので、係り受けの方向を予測する必要はない。したがって、我々のモデルを他の言語に適用する場合、これを予測するように拡張する必要がある。このモデルを用いた構文解析器を作成し、約 1,000 文の日本経済新聞からなるコーパスからパラメータを推定し、約 100 文に対して構文解析実験を行なった結果、係り受け単位で 89.9% の解析精度を得た。

2 係り受けに基づく確率的言語モデル

この節では、我々が提案する係り受けに基づく確率的言語モデルについて述べる。係り受けを記述する多くの確率的言語モデルと異なり、我々のモデルは隠れマルコフモデルの一つである。我々のモデルでは、文を構成するそれぞれの形態素が文頭から順に予測される。予測の各段階での履歴は、基本的には、係り先が未定の形態素の列である。構文に対する心理言語学的研究 [5] によれば、文の各位置において、係り先が未定の形態素の数には上限がある。この上限は短期記憶のためのスロットの数によって規定され、7±2 程度であるとされる [6]。この制限を用いることで、有限状態機械に基づく確率的言語モデルを作成することが可能となる。

2.1 文のモデル

我々が提案するモデルの基本的なアイデアは、それぞれの形態素の予測においてより重要な情報は、直前の形態素列 (形態素 n-gram モデルなど) ではなく、予測される形態素と係り受け関係にある形態素であるとの直観である。例として、図 1に示される文構造と図 2に示される 6 番目の形態素「りんご」が予測される過程の文構造の仮説について考察する。図 2の上の部分解析木は、1 つのノードだけの木 (m_3 からなる t_b) と、2 つのノードからなる木 (m_1 と m_2 からなる t_a、m_4 と m_5 からなる t_c) がある。仮に最後の 2 つの木 (t_b と t_c) が次の形態素 (m_6) に係るとすると、この形態素はこれらの形態素から予測されるのがよいであろう。このような観点から、我々のモデルは、まず、次の形態素に係る部分解析木を予測し、次いで、これらの木から次の形態素を予測する。

ここで、我々のモデルを形式的に説明するために以下の定義を行なう。

- $m = m_1 m_2 \cdots m_n$: 形態素列。形態素は文字列と品詞の対である。
- $t_i = t_1 t_2 \cdots t_{k_i}$: 先行する i 個の形態素を覆う部分解析木の列。
- t_i^+ 及び t_i^- : 次の形態素に係る部分解析木の列、及び次の形態素に係らない部分解析木の列。係り受け関係は交差しないと仮定しているので $t_i = t_i^- t_i^+$ である。
- $\langle t\, m \rangle$: m を根とし t を根の部分木とする木。形態素 (m_{i+1}) がそれに係る木の列 (t_i^+) から予測された後、係らない木の列 (t_i^-) と新たに生成された木 ($\langle t_i^+ m_{i+1} \rangle$) の連接が履歴となる。したがって $t_{i+1} = t_i^- \cdot \langle t_i^+ m_{i+1} \rangle$ である。
- y_{max} : 係り先が未定の形態素の数の上限。

以上の定義のもと、我々の確率的言語モデルは以下のよ

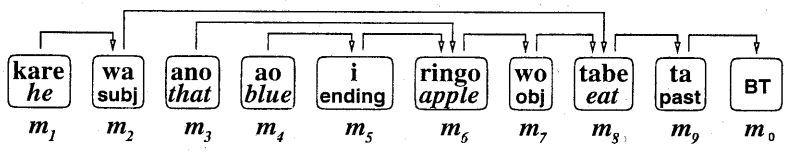

図 1: 文とその係り受け構造

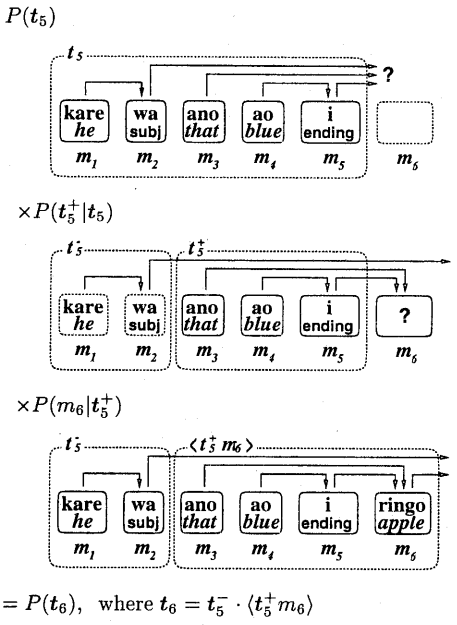

図 2: 部分解析からの形態素予測

うに定義される。

$$P(\boldsymbol{m}) = \prod_{i=1}^{n} P(m_i|m_1 m_2 \cdots m_{i-1}) \approx \sum_{\boldsymbol{t}_n \in T_n} \prod_{i=1}^{n} P(m_i|\boldsymbol{t}_{i-1}^{+}) P(\boldsymbol{t}_{i-1}^{+}|\boldsymbol{t}_{i-1}) \quad (1)$$

ここで、T_n は n ノードの可能なすべての木の集合を表す。この式の第 1 因子 ($P(m_i|\boldsymbol{t}_{i-1}^{+})$) を形態素予測モデルと呼び、第 2 因子 ($P(\boldsymbol{t}_{i-1}^{+}|\boldsymbol{t}_{i-1})$) を状態予測モデルと呼ぶ。もう一度図 2について考察する。上段の図は 6 番目の形態素の予測の直後の状態である。状態予測モデルは、まず次の形態素に係る部分解析木を予測し、中段の図となる。次いで形態素予測モデルが、次の形態素をそれに係る部分解析木から予測し、下段の図となる。

すでに述べたように、形態素予測の各段階において、係り先が未定の形態素の数には上限があると考えられる。つまり、部分解析木列 ($\boldsymbol{t}_i$) の要素数には上限がある。したがって、部分解析木の深さに上限があるとすると、可能なすべての状態の数は有限となる。この条件が満たされている限り、我々のモデルは隠れマルコフモデルである。隠れマルコフモデルにおいては、第 1 因子は出力確率と呼ばれ、第 2 因子は遷移確率と呼ばれる。

係り受け関係は交差しないことを仮定しているので、状態予測モデルは次の形態素に係る木の数を予測するだけで十分である。したがって、木の列 $\boldsymbol{t}_{i-1}^{+}$ の要素数を $y = |\boldsymbol{t}_{i-1}^{+}|$ として、$P(\boldsymbol{t}_{i-1}^{+}|\boldsymbol{t}_{i-1}) = P(y|\boldsymbol{t}_{i-1})$ となる。非交差の仮定から、最後の y 個の部分解析木が i 番目の形態素に係ることが分かる。形態素列に対する可能な解析木の数はその形態素数に対して指数関数的に増加するので、部分解析木の列の長さに上限がある場合でも、部分解析木の列がなす空間は広大である。このことは、データスパースネスの問題を引き起こす。この問題を避けるために、部分解析木の区別に利用するノードの深さを制限することとする。4節で述べる実験では、根とその子ノードのみを区別の対象とする。このように、最初のレベルの形態素と次のレベルの形態素を区別の対象とするモデルを P_{LL} と表す。したがって、実験に用いたモデルでは、次の形態素に係る木の数と形態素は、先行する形態素列を覆う部分解析木の深さ 2 以下の部分木を履歴と見なして予測される。もし、それぞれの形態素が次の形態素に係るという文構造を仮定すれば、我々のモデルは形態素 3-gram モデルと等価であることを付言しておく。

我々の提案するモデルを未知の入力に対して頑強にするために、n-gram モデルと同様の補間 [7] を行なう。木を区別する際の規則を緩和することで、より一般的なモデルが得られる。例えば、根とその子ノードの品詞のみを区別するとすれば品詞 3-gram モデルに似たモデル (以下 P_{PP} と表記) が得られる。根の形態素のみを区別し、その子ノードを無視すると形態素 2-gram モデルに似たモデル (以下 P_{NL} と表記) が得られる。平滑化の一手法として、形態素 3-gram に類似する P_{LL} を P_{PP} や P_{NL} などのより一般的なモデルと補間することが考えられる。実

験では、以下の式が示すように一般化のレベルが異なる7つのモデルを補間した。

$$\begin{aligned} P(m_i|t_{i-1}^+) &= \lambda_6 P_{LL}(m_i|t_{i-1}^+) + \lambda_5 P_{PL}(m_i|t_{i-1}^+) \\ &+\lambda_4 P_{PP}(m_i|t_{i-1}^+) + \lambda_3 P_{NL}(m_i|t_{i-1}^+) \\ &+\lambda_2 P_{NP}(m_i|t_{i-1}^+) + \lambda_1 P_{NN}(m_i|t_{i-1}^+) \\ &+\lambda_0 P_{m,0-\mathrm{gram}} \end{aligned} \quad (2)$$

ここで、P_{YX} の X は深さ1のノードの区別のレベル (N: 区別しない、P: 品詞、L: 形態素) を示し、Y は深さ2のノードの区別のレベルを示す。また $P_{m,0-\mathrm{gram}}$ は、語彙 $\mathcal{M}$ に対する一様分布を表す ($P_{m,0-\mathrm{gram}} = 1/|\mathcal{M}|$)。

状態予測モデル ($P(y|t_{i-1})$) も全く同様に補間される。この場合、可能な事象は $y = 1, 2, \ldots, y_{max}$ であるので、$P_{y,0-\mathrm{gram}} = 1/y_{max}$ である。

2.2 パラメータ推定

我々の提案するモデルは隠れマルコフモデルであるから、生コーパスから EM アルゴリズム [8] を用いてパラメータを推定することができる。このアルゴリズムを用いれば、生コーパスの出現確率が最大となるパラメータが推定される。この際、各文の構造は考慮されないので、結果として得られるモデルは、必ずしも構文解析に適切であるとは限らない。

構文解析に適切なモデルを構築することを目的とした場合、構文構造が付与されたコーパスから、以下の式が示すような、相対頻度による最尤推定を用いてパラメータを推定することが望ましい [3]。

$$P(m|t^+) \overset{\mathrm{MLE}}{=} \frac{f(\langle t^+ m_i \rangle)}{\sum_m f(\langle t^+ m_i \rangle)}$$

$$P(y|t) \overset{\mathrm{MLE}}{=} \frac{f(y,t)}{f(t)}, \quad \text{where } y = |t^+|$$

ここで、$f(x)$ は事象 x の学習コーパスでの頻度を表す。

式 (2) の補間係数は、削除補間法 [7] によって推定される。

2.3 語彙化する形態素の選択

一般的に、形態素 n-gram モデルは品詞 n-gram モデルよりも予測力が高い。しかしながら、低頻度の形態素を語彙化することは、しばしばデータスパースネスの問題を引き起こし、モデルに悪影響を及ぼす恐れがある。例えば、英語の形態素解析器 [9] では、頻度 100 以上の単語のみが語彙化されている。同様に、最も精度が高いとされる英語の構文解析器の一つ [10] では、学習コーパスに5回以上出現する単語のみが語彙化される。このような理由から、我々のモデルでは、語彙化する形態素を学習の時点で選択することとした。上述の形態素を区別の対象とするモデル (P_{LL} と P_{PL} と P_{NL}) では、選択された形態素のみを区別の対象とし、それ以外は品詞のみ区別する。選択の基準は、パラメータ推定とは別に用意された学習コーパスの一部であるヘルドアウトコーパスに対する構文解析の精度 (4節参照) とした。したがって、テストコーパスや未知の文の構文解析の精度を向上させると推定される形態素のみが語彙化される。この際のアルゴリズムは、以下の通りである (図3参照)。

1. 初期状態では形態素はそれぞれの品詞に対応するクラスに属している。
2. すべての形態素は頻度の降順に整列され、この順に以下の処理が実行される。
   (a) 注目する形態素を仮に語彙化して、ヘルドアウトコーパスの解析精度を計算する。
   (b) もし、精度の向上が観測されればこの形態素を語彙化する。

このアルゴリズムの計算結果を部分解析木の区別に用いる。つまり、木の区別に際しては語彙化された形態素のみの文字列をチェックし、語彙化されなかった形態素は品詞のみ区別する。もし仮に、語彙化される形態素がなければ、$P_{NL} = P_{NP}$ であり、$P_{LL} = P_{PL} = P_{PP}$ となる。形態素の併合も試みることとすれば、このアルゴリズムはトップダウンクラスタリングのアルゴリズムとなる。

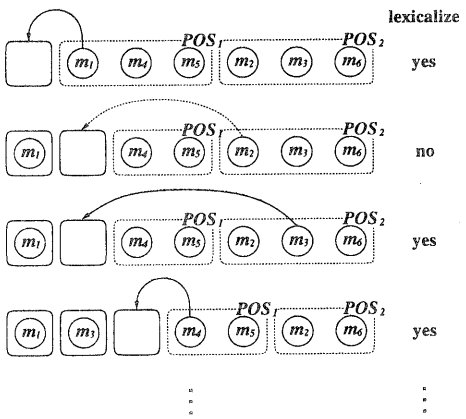

図 3: 語彙化のアルゴリズム

2.4 未知語モデル

我々の言語モデルが未知形態素を扱うことができるように、文字 2-gram からなる未知語モデルを付加した。語彙にない形態素を予測する場合、まずその品詞を予測し、次いで未知語モデルが品詞から文字列を以下の式を用いて予

測する。

$$P(m|POS) = \prod_{i=1}^{m+1} P_{POS}(x_i|x_{i-1})$$
$$\text{where } m = x_1 x_2 \cdots x_m,\ x_0 = x_{m+1} = \mathrm{BT}$$

ここで、BT は形態素の境界を示す特別の記号である。これにより、すべての可能な文字列に対する確率値の合計が 1 になる。

削除補間による補間係数の推定では、学習コーパスを複数の部分コーパスに分割する。実験では、語彙は部分コーパスの 2 つ以上に出現する形態素とした。未知語モデルのパラメータは、1 つの部分コーパスにしか出現しない形態素から、品詞毎に以下の式を用いて推定される。

$$P_{\mathrm{POS}}(x_i|x_{i-1}) \overset{\mathrm{MLE}}{=} \frac{f_{\mathrm{POS}}(x_i, x_{i-1})}{f_{\mathrm{POS}}(x_{i-1})}$$

文字 2-gram モデルも文字 1-gram モデルと文字 0-gram モデル (一様分布) と補間する。補間係数は、削除補間法 [7] により推定する。

3 構文解析

一般的に、従来の構文解析器は形態素列に分割された文を入力とし、その構造を出力する。これ対して、我々が提案する構文解析器は、未知語モデルを備えているので、文字列を入力として、単語への分割と品詞の付与を構文解析と同時に行なうことができる。この節では、前節で述べた言語モデルに基づく構文解析器について説明する。

3.1 確率的構文解析器

確率的言語モデルに基づく構文解析器は、文字列 ($\boldsymbol{x}$) を与えられると、確率が最大となる構造 (図 1参照) を以下の式に従って出力する。

$$\begin{aligned}
\hat{T} &= \operatorname*{\mathbf{argmax}}_{\boldsymbol{m}(T)=\boldsymbol{x}} P(T|\boldsymbol{x}) \\
&= \operatorname*{\mathbf{argmax}}_{\boldsymbol{m}(T)=\boldsymbol{x}} P(T|\boldsymbol{x})P(\boldsymbol{x}) \\
&= \operatorname*{\mathbf{argmax}}_{\boldsymbol{m}(T)=\boldsymbol{x}} P(\boldsymbol{x}|T)P(T) \quad (\because \text{Bayes' formula}) \\
&= \operatorname*{\mathbf{argmax}}_{\boldsymbol{m}(T)=\boldsymbol{x}} P(T) \quad (\because P(\boldsymbol{x}|T) = 1),
\end{aligned}$$

ここで、$\boldsymbol{m}(T)$ は構文木 T の形態素列の表記の連接である。最後の行の $P(T)$ は確率的言語モデルである。これは、我々の構文解析器では、2節で述べた確率的言語モデルによって計算される構文解析木 T の確率である。

$$P(T) = \prod_{i=1}^{n} P(m_i|\boldsymbol{t}_{i-1}^{+})P(\boldsymbol{t}_{i-1}^{+}|\boldsymbol{t}_{i-1}) \qquad (3)$$

3.2 解探索のアルゴリズム

式 (3) に示されるように、我々の構文解析器は隠れマルコフモデルに基づいている。したがって、最適解の探索には Viterbi アルゴリズムが適用できる。Viterbi アルゴリズムは、入力の文字数を n とすると、$O(n)$ の時間で最適の解を計算することができる。

構文解析器は、入力の文字列を順に読み込みながら状態遷移を繰り返す。出力される構造は 1 つの木でなければならない。そのためには、最終状態 $\boldsymbol{t}_n$ の木の数はちょうど 1 である必要がある。構文解析器は、この条件を満たす最終状態の中から、最大の確率となる状態を選ぶ。提案する言語モデルでは、根とその子ノードのみで部分木を区別するので、最終状態の情報だけでは構文木を構成することができない。しかしながら、構文木は、状態の列、つまり形態素の列と係り先が未定の形態素の列から構成することができる。したがって、構文解析器は、予測の各段階でこれらを記憶し、確率最大の最終状態が選択されると、これらの列を順に読み構文木を構成する。

表 1: コーパス

	文数	形態素数	文字数
学習	1,072	30,292	46,212
テスト	119	3,268	4,909

4 評価

2節で述べた、品詞に基づくモデルとそれを語彙化したモデルを構成し、それぞれの予測力を計算した。さらに、これらの言語モデルに基づき、3節で説明した解探索アルゴリズムを用いる構文解析器を作成し、テストコーパスに対する構文解析の実験を行なった。この節では、この結果を提示し、その評価を行なう。

4.1 実験の条件

実験に用いたのは、日本経済新聞の記事に含まれる文からなるコーパスである。各文は、形態素に分割され、構文構造が付与されている。この作業は、専用に設計したエディターを用いて、人手で行なった。コーパスは 11 個に分割され、この内の 10 個を学習コーパスとしモデルの構成に用い、残りの 1 個をテストコーパスとした (表 1参照)。学習コーパスの内の 1 個は 2節で述べた語彙化アルゴリズムにおけるヘルドアウトコーパスとした。予測の各時点で係り先が未定の形態素の数を学習コーパスについて調べた結果、この値を 10 とすることとした ($y_{max} = 10$)。

予測力を評価する目的で、各モデルのテストコーパスに対するクロスエントロピーを計算した。この計算を行なう際には、コーパスに付加された構造を用いた。したがって、すべての可能な導出方法の確率の和を用いているのではない。品詞に基づくモデルと語彙化したモデルを比較す

表 2: クロスエントロピーと解析精度

言語モデル	クロス エントロピー	解析精度
選択的語彙化モデル	6.927	89.9%
盲目的語彙化モデル	6.651	87.1%
品詞に基づくモデル	7.000	87.5%
ベースライン *	—	78.7%

* それぞれの形態素は次の形態素に係るとする

るために、それぞれを同じ学習コーパスから推定し、同じテストコーパスに対するクロスエントロピーを計算した。未知語モデルは共有なので、この部分のクロスエントロピーへの寄与は一定である。

次に、係り受けモデルによる構文解析器を作成し構文解析の精度の評価を行なった。構文解析器は、未知語モデルを備えているので、文字列に対する構文解析も可能であるが、形態素への分割の誤りが生じ、結果の評価が困難であるため、形態素列を入力とすることとした。

構文解析結果の評価基準は出力される係り受け関係の精度である。これは、日本語の係り受け解析の標準的な評価方法である。精度は、以下の式で示されるように、基本的には推定した係り受け関係の数 (形態素の数に等しい) に対する、正しい係り受け関係の割合である。ここで、正しい係り受け関係とは、コーパスに付与された係り受け関係と同じでることを意味する。

$$\text{解析精度} = \frac{\text{係り先が正しい形態素の数}}{\text{形態素の数}}$$

ただし、最後の形態素と最後から 2 番目の形態素は、評価の対象としない。この理由は、最後の形態素には係り先がなく、最後から 2 番目の形態素は、必ず最後の形態素に係るので、曖昧性がないことである。

4.2 評価

表 2は、1) すべての形態素が次の形態素に係るとするベースライン、2) すべての形態素を語彙化したモデル、3) 品詞に基づくモデル、4)2節で示したアルゴリズムにより選択的に語彙化したモデルによる構文解析の精度である。この結果から、選択的に語彙化したモデルに基づく構文解析器の精度が最も高いことがわかる。しかしながら、クロスエントロピーが最小であるのは、すべての形態素を語彙化したモデルである。このことから、音声認識などの言語モデルとしては、このモデルが最良であると推定される。クロスエントロピーと音声認識の精度には、間接的な関係があると報告されているので、音声認識と構文解析を同時に行なうための言語モデルとしては、語彙化する形態素の選択に他の基準を用いることも考えるべきであろう。

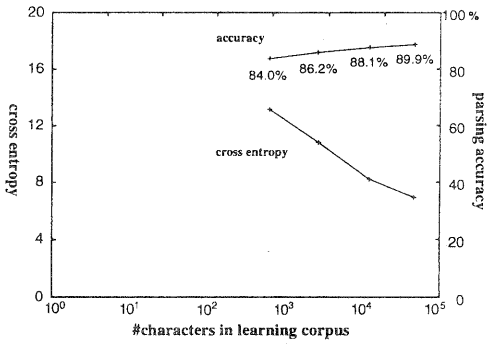

図 4: クロスエントロピーと解析精度の関係

次に、学習コーパスの大きさに対するクロスエントロピーと解析精度の変化を調べた結果について述べる。図4は、テストコーパスを一定保ったまま、学習コーパスの大きさをそれぞれ 1/4、1/16、1/64 とした場合のクロスエントロピーと解析精度である。クロスエントロピーは、学習コーパスの増加に従って強く減少する傾向があることがわかる。解析精度にも同様に、学習コーパスの増加に従って向上する傾向がある。この結果から、現段階では学習コーパスを増加させることで、より高い精度の構文解析器が得られることが分かる。また、他の構文解析器との比較という点では、実験に用いたコーパスサイズが非常に小さいことを考えると、我々の構文解析器の精度は、現時点での最高レベルにあり、我々のアプローチが効果的であると結論できる。

5 関連研究

歴史的には、自然言語の構造の記述には文脈自由文法が用いられ、構造的な曖昧性の解消には主としてこの文法に基づく構文解析器 [11] が用いられてきた。これに対して、最近では、有限状態モデルを用いた構文解析の研究がいくつかなされている [12]。我々の構文解析器も有限状態モデルに基づいているが、人間の記憶能力に関する制限 [5, 6] に基づいている点で異なる。したがって、我々のモデルは心理言語学的により適切であろう。

確率文脈自由文法の分野での最近の潮流として、語彙の重要視があげられる [10, 13]。これらの文献では、構文解析器を語彙化することにより解析精度の有意な向上が見られたと報告されている。このような知見を考慮して、本論文では語彙化する形態素を解析精度などの一定の基準で選択することを提案し、これにより有意な精度の向上が見られることを報告した。我々が提案する語彙化の方法は、確率文脈自由文法に基づく構文解析器にも適用可能であり、

精度向上に貢献するであろう。

本論文で提案したモデルは、生成的な確率言語モデルである。この分野では、Chelba と Jelinek [14] が類似するモデルを提案している。彼らのモデルでは、係り受け関係の有無に関わらず、それぞれの単語は最も右に位置する2つの部分解析木の主辞から予測される。このモデルは、部分解析木の内の2つが常に次の単語に係る場合、我々のモデルとほぼ等価である。他に、確率文脈自由文法を語彙化したモデルが提案されている [15]。このモデルは、最高水準にある英語の構文解析器 [16] の言語モデルと同じように、それぞれの単語は、それを含む句の主辞と隣接する単語の主辞に近い方から予測される。我々の提案するモデルとこれらのモデルとの最大の相違点は、係り受け関係にある部分解析木を用いることである。これは、心理言語学的により適切であろう。他の相違点として、我々のモデルは隠れマルコフモデルの1つであるので、入力の長さを n とした場合の計算時間が $O(n)$ であり、文脈自由文法に基づくモデルの $O(n^3)$ よりも高速である点が挙げられよう。

日本語の構文解析の研究としてもいくつかの先行研究がある [17, 18, 19, 20]。これらの構文解析器は文節に基づいており、文節列を入力として、それらの係り受け関係を出力する。これらの構文解析器と異なり、我々が提案するモデルは、形態素間の係り受け関係を記述する。文節という単位をモデルに含めなかった理由は、文節の定義が曖昧であることと、文節が後続する処理に必ずしも必要ではないことである。構文解析の精度という点では、比較は容易ではないが、我々の構文解析器 (1,072 文で学習、89.9% の精度) は、上述の他の構文解析器 (50,000 ～ 190,000 文で学習、82% ～ 85% の精度) と同等かそれ以上の性能であると考える。記述能力を比較した場合、従来のモデルが2項関係のみをモデル化しているのに対し、我々のモデルは3項以上の係り受け関係をモデル化している。これは、格フレームなどの記述 (図1における「は/助詞」と「を/助詞」からの「食べ/動詞」の予測) や、先行する係り受け関係による振る舞いの変化 (詳細化) の記述 (図1における「を/助詞」からの「食べ/動詞」の予測において「を/助詞」に「林檎/名詞」が係っていることを考慮すること) に必要である。この点において、我々の言語モデルは他のモデルに対して優れている。

6 結論

本論文では、係り受け構造を基礎とする確率的言語モデルについて述べた。このモデルでは、文は形態素の列とみなされ、それぞれの形態素は文頭から順に予測される。予測の各段階での履歴は、その時点までの形態素列を覆う部分解析木の列である。このモデルでは、まず、次の形態素に係る部分解析木を予測し、それから、次の形態素をそれに係る部分解析木から予測する。モデルを改善する方法として、解析精度を向上させると予測される形態素のみを選択的に語彙化するアルゴリズムについても述べた。このモデルを用いた構文解析器を作成し、日本経済新聞からならる約 1,000 文のコーパスからパラメータを推定し、約 100 文に対して構文解析実験を行なった結果、係り受け単位で 89.9% の解析精度を得た。

参考文献

[1] Kenneth Ward Church. A Stochastic Parts Program and Noun Phrase Parser for Unrestricted Text. In *Proceedings of the Second Conference on Applied Natural Language Processing*, pp. 136–143, 1988.

[2] Doug Cutting, Julian Kupiec, Jan Pedersen, and Penelope Sibun. A Practical Part-of-Speech Tagger. In *Proceedings of the Third Conference on Applied Natural Language Processing*, pp. 133–140, 1992.

[3] Bernard Merialdo. Tagging English Text with a Probabilistic Model. *Computational Linguistics*, Vol. 20, No. 2, pp. 155–171, 1994.

[4] Evangelos Dermatas and George Kokkinakis. Automatic Stochastic Tagging of Natural Language Texts. *Computational Linguistics*, Vol. 21, No. 2, pp. 137–163, 1995.

[5] Victor H. Yngve. A Model and a Hypothesis for Language Structure. *The American Philosophical Society*, Vol. 104, No. 5, pp. 444–466, 1960.

[6] George A. Miller. The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. *The Psychological Review*, Vol. 63, pp. 81–97, 1956.

[7] Fredelick Jelinek, Robert L. Mercer, and Salim Roukos. Principles of Lexical Language Modeling for Speech Recognition. In *Advances in Speech Signal Processing*, chapter 21, pp. 651–699. Dekker, 1991.

[8] L. E. Baum. An Inequality and Associated Maximization Technique in Statistical Estimation for Probabilistic Functions of Markov Process. *Inequalities*, Vol. 3, pp. 1–8, 1972.

[9] Julian Kupiec. Augmenting a Hidden Markov Model for Phrase-Dependent Word Tagging. In

Proceedings of the DARPA Speech and Natural Language Workshop, pp. 92–98, 1989.

[10] Michael Collins. Three Generative, Lexicalised Models for Statistical Parsing. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, pp. 16–23, 1997.

[11] T. Fujisaki, F. Jelinek, J. Cocke, E. Black, and T. Nishino. A Probabilistic Parsing Method for Sentence Disambiguation. In *Proceedings of the International Parsing Workshop*, 1989.

[12] Kemal Oflazer. Dependency Parsing with an Extended Finite State Approach. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pp. 254–260, 1999.

[13] Eugene Charniak. Statistical Parsing with a Context-free Grammar and Word Statistics. In *Proceedings of the 14th National Conference on Artificial Intelligence*, pp. 598–603, 1997.

[14] Ciprian Chelba and Frederic Jelinek. Exploiting Syntactic Structure for Language Modeling. In *Proceedings of the 17th International Conference on Computational Linguistics*, pp. 225–231, 1998.

[15] Jason M. Eisner. Three New Probabilistic Models for Dependency Parsing: An Exploration. In *Proceedings of the 16th International Conference on Computational Linguistics*, pp. 340–345, 1996.

[16] Michael John Collins. A New Statistical Parser Based on Bigram Lexical Dependencies. In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, pp. 184–191, 1996.

[17] Masahiko Haruno, Satoshi Shirai, and Yoshifumi Ooyama. Using Decision Trees to Construct a Practical Parser. In *Proceedings of the 17th International Conference on Computational Linguistics*, pp. 505–511, 1998.

[18] Masakazu Fujio and Yuji Matsumoto. Japanese Dependency Structure Analysis based on Lexicalized Statistics. In *Proceedings of the Third Conference on Empirical Methods in Natural Language Processing*, pp. 87–96, 1998.

[19] Kiyoaki Shirai, Kentaro Inui, Hozumi Tanaka, and Takenobu Tokunaga. An empirical study on statiscical disambiguation of Japanese dependency structures using a lexically sensitive language model. In *NLPRS*, pp. 215–220, 1997.

[20] Shinsuke Mori and Makoto Nagao. A Stochastic Language Model using Dependency and Its Improvement by Word Clustering. In *Proceedings of the 17th International Conference on Computational Linguistics*, pp. 898–904, 1998.