情 報 学 基 礎 61−13
自 然 言 語 処 理 142−13
（2001. 3. 5）

# BNNを用いた日本語文の係り受け解析

長田 靖　　吉田 敬一

静岡大学大学院理工学研究科

**概要**

本研究は、係り受け候補を求めるにあたって、コーパスを用いることで出来る限り人手をかけずに係り受け候補を求めるものである。係り受けの妥当性の判定には、正しい係り受けのとき評価値が最小となるように評価関数を定義した。その際に、コーパスから得られる係り受けの頻度の情報を加えることにより、文節（単語）間の意味的関係を考慮できるよう工夫した。最小値検索にはボルツマン・ニューラル・ネットワーク (Boltzmann Neural Network;BNN) を用いた。コーパスを利用することにより、人手のかからない解析システムを構築し、精度を向上させることを目的とした。

# Dependency-based Analysis for Japanese Sentences with Boltzmann Neural Networks

Osamu Nagata and Keiichi Yoshida

Graduate School of Science and Engineering, Shizuoka University

**Abstract**

In this paper, we propose a method that we find most proper dependency-based model, with little hand-process,using a corpus. We defined such a function that it has a minimum value when our model were found. In this function definning, we included the frequency of each bunsetu phrase pair appeared in the corpus. We apply this function to verify whether the bunsetu phrase pair, in a given sentence is fit or not. We apply Boltzmann Neural Networks for finding a minimum value in the function. We implemented a powerful dependency analysis model by using the method described above.

# 1 はじめに

構文解析の一つとして、文節間の係り受けを明らかにする係り受け解析がある。一般的に係り受け解析では、係り受け可能な二つの文節のタイプを分類したものを文法規則として人手で与えている。しかし、これには大変な労力が伴う。近年、大規模な文書データであるコーパスが普及しはじめ、構文解析済みのデータが付属しているコーパスがある。本研究では、この構文解析済みコーパスを用いてその中に存在する係り受けを、入力文の係り受け候補とし、解析を進める。
また、係り受けの妥当性の判定に際しては、一般的によく知られている「非交差性」、「係り先占有性」、「卑近接続性」がある。さらに、本研究では精度をあげるためコーパスから求められる係り受けの頻度の情報も用いた。
評価値の最小値検索にボルツマン・ニューラル・ネットワーク (Boltzmann Neural Network;BNN) を用いた。これらを用いた本研究での解析システムについて報告する。

# 2 係り受け解析

係り受け解析とは、文節単位に区切られた日本語文に対し、考えられる全ての係り受け (図 1 の実線及び破線矢印) の中から、正しい係り受け (図 1 の実線矢印) を選び出すことである。ここで係り受け線 $i$ を、それが選択されていれば値"1"、そうでなければ値"0"を持つ変数 $u_i$ で表す。この時、図 1 の例で妥当な係り受け表すベクトルは以下のようになる。

$$u = [1 \quad 0 \quad 1 \quad 0 \quad 1 \quad 0 \quad 1 \quad 1]$$

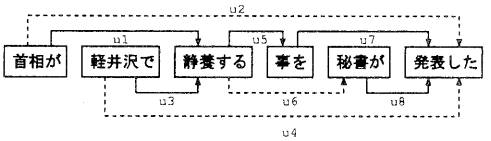


図 1: 文節間係り受け関係の例

# 3 ニューラルネットワーク

人間の脳の仕組みをモデル化したもので、今までに様々なモデルが提案されている [3, 4, 5]。それらは大きく二つに分類する事ができる。一つは、情報の流れが一方向に決まっているもので、これを階層型と呼ぶ。もう一つは情報の流れが双方向なもの、これを相互結合型と呼ぶ。本研究で用いるのは相互結合型のニューラルネットワークの一つである BNN である。

## 3.1 Hopfield のネットワーク

相互結合型のニューラルネットワークの一つに、Hopfield のネットワークがある。このモデルでは、次の二つの制約条件を用いている。

1 ユニット間の相互結合の強さは対象である。
$w_{ij} = w_{ji}$

2 各ユニットは非同期的に動作する。

ここで、$w_{ij}$ はユニット $j$ からユニット $i$ への結合の強さ (重み) を表す。N 個のユニットからなるネットワークにおいて、時刻 $t$ に $i$ 番目のユニットが $N-1$ 個のユニットから受け取る信号の総和 $u_i(t)$ を、以下の式で表す。

$$u_i(t) = \sum_{i=1}^{n} w_{ij} v_j(t) + \theta_i \tag{1}$$

ここで $v_j(t)$ は時刻 $t$ におけるユニット $j$ の出力で、$\theta_i$ はユニット $i$ の閾値である。
そして次に $u_i(t)$ の値によって時刻 $t+1$ のユニット $i$ の出力を次の式に従って変化させる。

$$v_i(t+1) = \begin{cases} 1 & u_i(t) \geq 0 \\ 0 & u_i(t) < 0 \end{cases} \tag{2}$$

このとき、$i$ 以外のユニット $j$ の出力は変化させない。

ネットワークの状態が一つ決まると、そのネットワークのエネルギーは次式で定義される。

$$E = -\frac{1}{2} \sum_{i=1}^{n} \sum_{j \neq i} w_{ij} v_i v_j + \sum_{i=1}^{n} \theta_i v_i \tag{3}$$

エネルギー $E$ は状態遷移が進むにつれて必ず減少する (または変化しない) 事が証明されている [3, 4, 5]。そのため、ある問題を表す評価関数の値が、正しい解ほど数値が少なくなるように設定し、その評価関数をネットワークのエネルギー-$E$ の式と等価であるとみなすことにより、組み合わせ最適化問題を解くことができる。

しかし、Hopfield のネットワークは常にエネルギーが減少するように状態遷移するため、初期値により必ずしもエネルギー最小状態にたどり着かずに、エネルギー極小状態に収束する可能性がある。これを防ぐために確率を導入したのが BNN(Boltzmann Neural Network) である。

### 3.2 BNN

BNN はユニットの出力関数として次式を用いる。それ以外は Hopfield のネットワークと同じである。
Hopfield のネットワークでは入力の総和が正か負かで出力を一意に決定していたが、BNN では出力が"1"になる確率を次式で定義する。

$$p[v_i(t+1)=1] = \frac{1}{1+\exp\left(-\frac{u_i(t)}{T}\right)} \qquad (4)$$

ここで $T$ はネットワークの温度と呼ばれる正の定数で、温度が高いほど図 2 のように応答曲線はなだらかになり、$T\to\infty$ の極限で入力の総和に関係なく出力が 1 になる確率が 0.5 になり、$T\to 0$ の極限で Hopfield のネットワークの出力関数と等しくなる。
入力の総和が負であっても低い確率で 1 を出力する事を許し、これによりエネルギー増加の方向への状態遷移を許したため、エネルギー極小状態からの脱出を可能にしている。

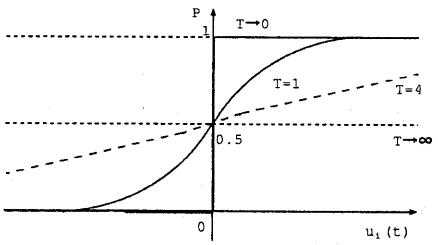


図 2: 式 (4) のグラフ

## 4 提案する手法

一般的な係り受け解析の手法では、入力文中の係り受け候補の探索においては係り受け可能な二つの文節のタイプを分類 (表 1 参照) し、それをもとに取りうる全ての係り受けを求める。しかし、こうした方法による分類パターン (係り受け規則) の作成には大変な労力を伴う。そこで、本研究ではコーパスを用いてトレーニングを行い、トレーニングデータの中に出現した文節間の係り受けを用いて、入力文の中に存在しうる全ての係り受けを求める。そして、それらの中から最も妥当な組合せを求めるため、以下の構文原理を仮定し、評価関数を定義し、BNN で最小値検索することで解析を進めて行く。

表 1: 係り受け文法の例

| 係る文節のタイプ | 受ける文節のタイプ |
|---|---|
| 名詞+格助詞「が」 | 動詞 (ガ格の格要素を支配する動詞) |
| 名詞+格助詞「を」 | 動詞 (ヲ格の格要素を支配する動詞) |
| ... | ... |
| 動詞 連用形 | 動詞, 形容詞, 形容動詞 |
| 動詞 連体形 | 名詞 |
| ... | ... |

### 4.1 構文原理

本研究では次のような係り受けの性質を仮定し、それぞれに対する評価関数を定義して解析に用いる。

1 妥当な解釈においては係り受け線は交差しない (非交差性)

2 ある文節の係り先は一つである。(係り先占有性)

3 距離的に近い文節ほど係り受けが成立しやすい。(卑近接続性)

4 コーパスから求められた頻度の高い係り受けほど妥当である。(頻度との関連性)

1 は書き言葉ではほとんどの場合成り立つため、絶対的な規則 (制約) として用いられることが多い。2 は実際の係り受けを考えた場合、当然の条件である。3 は経験的に有効な優先規則であることが知られている。4 については、係り受けが妥当であるか否かの判定には係り受けの頻度が大きく影響を与えると考えたため、この条件を追加した。コーパスに多く現われる係り受けは意味的にも構文的にも正しいので、それらの係り受けは優先すべきであると考えた。実際の文章の中に

現われる係り受けの頻度(確率)を導入する事により、単に人手のコストを抑制するだけでなく、文節(単語)間の意味的関係についても考慮する事ができ、さらなる解析精度の向上にもつながると考える。

## 4.2 評価関数

前節で示した 構文原理1、2、3、4を表す評価関数をそれぞれ $E_1$,$E_2$,$E_3$,$E_4$ とする。そしてそれぞれを以下のように定義する。

### 4.2.1 非交差性 $E_1$

$$E_1 = \sum_{i=1}^{n}\sum_{j\neq i} X_{ij}u_iu_j \tag{5}$$

ただし、

$$X_{ij} = \begin{cases} 1 & \text{係り受け関係 } i \text{ と } j \text{ が交差している場合} \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

ある係り受け $i$,$j$ が交差しているごとに2ずつ増加。正解時には0となる。

### 4.2.2 係り先占有性 $E_2$

$$E_2 = \sum_{k=1}^{m}(\sum_{i=1}^{n} Y_{ki}u_i - 1)^2 - 1 \tag{7}$$

ただし、

$$Y_{ki} = \begin{cases} 1 & \text{係り受け関係 } i \text{ の係り元が文節 } k \text{ である場合} \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

係り受け線の過不足の2乗和。正解時(それぞれ一つの文節から一本ずつしか係り受け線が出ていなければ)には0となる。

### 4.2.3 卑近接続性 $E_3$

$$E_3 = \sum_{i=1}^{n} Z_iu_i \tag{9}$$

ただし、

$$Z_i = l_i - k_i \tag{10}$$

ここで、$l_i$、$k_i$ は、それぞれ係り受け $i$ の係り元文節番号、係り先文節番号を表す。
この評価値は正解時にも最小になるとは限らない。

### 4.2.4 係り受けの頻度 $E_4$

$$E_4 = \sum_{i=1}^{n} Q_iu_i \tag{11}$$

ただし、

$$Q_i = \begin{cases} 0 & h_i = 0 \\ -\frac{h_i}{h_{max}} & otherwise \end{cases} \tag{12}$$

$h_i$ コーパスから求めた係り受け $i$ の頻度

$h_{max} = max\{h_i|i = 1,\ldots,n\}$

入力文によって現われる係り受けも異なるので、入力文によって頻度の値も大きく代わってくるだろう。そのため、頻度そのものや頻度の逆数をパラメータとして用いると、場合によっては値が大きすぎたり、逆に小さすぎたりして、システムの振舞を狂わせるおそれがある。それでは安定したシステムを構築できるとは思えない。そこで、入力文の中に出てくる係り受けの最大の頻度のもので割ることにより、$Q_i$ の値が0から1の間に収まるように設定した。頻度が0、つまりコーパス中に一度も現われなかった場合を0とし、これを基準としたため、頻度が高いものほど低い値を示すように、符号を反転させた。

## 4.3 評価関数の最小値検索

以上4つの評価関数に対してその和をとることで、構文原理を反映するような評価関数を次のように定義する。

$$E = aE_1 + bE_2 + cE_3 + dE_4 \tag{13}$$

a,b,c,d は定数である。本研究ではこれらの定数を実験的に求める。
ニューラルネットを用いた係り受け解析では、各係り受け線にユニットを一つずつ割り当てる。この評価関数の式とBNNのエネルギーの式との対応をとることで、ネットワークの重み、閾値を与えるものとして以下の式が得られる。

$$w_{ij} = -2\{aX_{ij} + b\sum_{k=1}^{m} Y_{ki}Y_{kj}\} \tag{14}$$

$$\theta_i = -2b\sum_{k=1}^{m} Y_{ki} + cZ_i + dQ_i \tag{15}$$

これらを用いて係り受け解析用のネットワークを構築し、収束するまでネットワークを動かす。そして、収束した時の各ユニットの出力の組合せ(ネットワークの状態)を、与えられた入力文に対する解とする。

# 5 実験

ここまで説明した本研究の手法をを実際に実装し、実験を行った。以下にその結果について報告する。

## 5.1 トレーニング

実験に先立って、係り受け候補の探索、評価関数の計算に用いるための情報を EDR コーパスからトレーニングを行った。EDR コーパスは単語単位での依存構造が解析済みになっている。これを文節単位の依存構造に作り替え、コーパスに現れた係り受け、その頻度についてのデータを収集した。トレーニングには、ランダムに抜き出した１０万文を用いた。

表 2: トレーニングデータの例

| 係り側 | 受け側 | コーパスに現れた回数 |
|---|---|---|
| これを | もとに | 21 |
| これを | 受けて | 63 |
| ... | ... | ... |
| 熊本市内の | 小中学校では | 1 |
| 訓練が | 披露された | 3 |
| ... | ... | ... |

## 5.2 クーリングスケジュール

時刻 $t$ における温度 $T(t)$ をどの程度の速さで下げていくか（クーリングスケジュール）には、本研究では次式を用いた。

$$T(t) = \frac{T_0}{1+t} \tag{16}$$

ここで $T_0$ は初期温度をあらわす。
今回は初期温度 100 から $T(t)$ が 0.01 になるまで状態遷移を繰り返し、ネットワークを収束させた。

## 5.3 予備実験

評価関数の計算に用いる定数 a,b,c,d を求めるために、予備実験を行った。それぞれの値を 1 から 20 まで変化させながら（つまり $20^4$ 通りの組み合わせそれぞれについて）100 文で解析を行い、もっとも結果のよかった組み合わせを本実験で用いることにした。今回はこの方法で $a = 1, b = 20, c = 1, d = 20$ が求まった。

## 5.4 本実験

今回はクローズドテストとして、トレーニングに用いた文の中からランダムに 1000 文を抜き出し、上記の条件を元に実験を行った。実験結果に対する評価尺度としては、以下に示す適合率を用いた。

$$\text{適合率} = \frac{\text{正解データと一致した係り受けの数}}{\text{解析結果に含まれる係り受けの数}} \tag{17}$$

その結果、係り受けで 95.1%の解析精度が得られた。類似した研究 [2] では、学習用例文の係り受けと入力文の係り受けが類似している場合、解析精度 95%であったと報告されている。人手を大幅に抑制した本手法でも、人手で係り受け候補を求めるときとほぼ等しい精度が得られることが分かった。

# 6 おわりに

本研究ではコーパスを有効に活用することにより、なるべく人手がかからない係り受け解析システムを提案した。実際の実験ではクローズドテストにおいて、非常に高い精度が得られ、人手で与える規則無しにコーパスのみでの係り受け候補の探索、及び係り受けの頻度の情報を加えた本研究の解析システムの妥当性を示した。オープンテストにおいては、コーパスに現れない未知の係り受けが存在するため、データスパースの問題が生じてくると考えられる。しかし、トレーニングの量を増やしたり、未知の係り受けに対する対策を施すことで、同様によい結果が期待できると考える。引き続き、さらなる精度の向上のための改良、オープンテストに対して実験、考察を行っていきたい。

# 参考文献


[1] 清水 浩行, 佐藤 秀樹, 立岡 章, 林 達也 : ニューラルネットに基づく日本語係り受け解析, 情報処理学会研究報告 95-NL-108, pp.103-110,1995

[2] 高橋 直人, 板橋 秀一 : ニューラルネットワークを用いた日本語解析の試み, 情報処理学会論文誌 Vol.32 No.10, pp.1330-1337, 1991

[3] 中野 馨 [監修], 飯沼 一元 [編], ニューロンネットグループ, 桐谷 滋 [著] : 入門と実習 ニューロコンピュータ, 技術評論社, 1989

[4] 麻生 秀樹 著 : ニューラルネットワーク情報処理, 産業図書, 1988

[5] 松岡 清利 編著：ニューロコンピューティング，朝倉書店，1992

[6] 長尾 真 編：自然言語処理，岩波書店，1996