8N-3

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

1 はじめに

本研究は、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手法を提案するものである。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な評価関数を定義し、その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) を用いる。評価関数を設定する際に、コーパスから得られる係り受けの情報を加える事により、文節 (単語) 間の意味的関係を考慮できるようにした。これらにより、人手の介入を出来るだけ抑制した解析システムを構築し、精度を向上させるものである。

2 関連研究

BNNを用いた係り受け解析の研究に清水らの研究 [1] があり、清水らの研究では、まず意味的、文法的に存在しうるすべての係り受けを求め、以下で示す構文原理に基づいて評価関数を定義している。

1 妥当な解釈においては文節間の係り受けは互いに交差しない。

2 妥当な解釈においては、(最後の文節を除き) 全ての文節は自分より後方の文節に高々一つ係る。

3 距離的に近い文節ほど係り受けが成立しやすい。

4 一般に文中に複数個存在する用言 (動詞、形容詞、形容動詞) 等が必要とする必須格が満たされるような解釈が、妥当な解釈となる。

これら4つの項をもとに、彼らは評価関数を定義しているが、必須格 (第4項) が満たされているか判別するためには、用言が必要とする格のリストを人手で作成しなければならないため、人手のコストが増大するという欠点がある。

3 提案する手法

清水らの研究も含め一般的に、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) 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、3.1節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
...	...

3.1 構文原理

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、2、3においては清水らの研究と同じである。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については必須格が過不足ないか調べるのは人手がかかるため、本研究では条件から外した。また、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も大きく影響を与えると

考え、新しい4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る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)の情報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。

3.2 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。

3.2.1 非交差性 E_1

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

ただし、

$$X_{ij} = \begin{cases} 1 & \text{係り受け関係 } i \text{ と } j \text{ が交差している(相互排他的である)場合} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

3.2.2 係り先占有性 E_2

$$E_2 = \sum_{k=1}^{m} (\sum_{i=1}^{n} Y_{ki} u_i - 1)^2 - 1 \tag{3}$$

ただし、

$$Y_{ki} = \begin{cases} 1 & \text{係り受け関係 } i \text{ の係り元が文節(parse)} k \text{ である場合} \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

3.2.3 卑近接続性 E_3

$$E_3 = \sum_{i=1}^{n} Z_i u_i \tag{5}$$

ただし、

$$Z_i = l_i - k_i \tag{6}$$

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。

3.2.4 頻度との関連性 E_4

$$E_4' = \sum_{i=1}^{n} Q_i u_i \tag{7}$$

ただし、

$$Q_i = \begin{cases} 0 & h_i = 0 \\ -\frac{h_i}{h_{max}} & otherwise \end{cases} \tag{8}$$

h_i コーパスから求めた係り受け i の頻度

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

3.3 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、全ての構文原理を反映するような文の評価関数を次のように定義する。

$$E = aE_1 + bE_2 + cE_3 + dE_4 \tag{9}$$

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。

$$w_{ij} = -2\{aX_{ij} + b\sum_{k=1}^{m} Y_{ki} Y_{kj}\} \tag{10}$$

$$\theta_i = cZ_i - b - dQ_i \tag{11}$$

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。

4 おわりに

本研究ではコーパスを用いることにより、ほどんど人手のコストがかからない係り受け解析システムを提案した。本手法ではコーパスからの係り受け頻度の情報を文節レベルで考慮したことで、精度の向上にも期待ができると思われる。今後このシステムを実装し、その性能を評価し、、考察したい。

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] 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

[3] 長尾 真 編:自然言語処理, 岩波書店, 1996