統合コネクショニストモデルによる日本語複文解析システムの構築

本木 実　　嶋津好生
九州産業大学工学部
{motoki.shimazu}@te.kyusan-u.ac.jp

1. はじめに

近年，Elmanの研究を発端に，コネクショニストモデルによる自然言語理解研究が盛んに行われてきている．コネクショニストモデルは，学習により入出力間の写像を獲得し，未学習の入力に対しても適当な出力を行なう汎化能力を持つ．この獲得機構を自然言語処理に適用し，コネクショニストモデルが，人間の言語獲得の機能を説明できる認知モデルとしてのみならず，実用システムの一端を担えるかどうかが追求されている．

そのなかでも，MiikkulanenのSPEC[1]やJainのPARSEC[2]は，英語に関して，構文的に複雑な埋め込み文に対する格構造解析（'浅い'意味解析）を高い性能で実現している．これらは，コネクショニストモデルが構文解析のみ，あるいは単文レベルでの意味解析にとどまらず，複文という複雑な構造を持つ文の意味解析にも適用できる可能性を示すものである．

しかし，彼らのモデルは，英語という構文構造がある程度定まった言語を対象としたモデルであり，日本語のような文節の倒置が可能な，そして述語動詞が文の最後にくるSOV型の言語に対するモデルはまだない．そこでわれわれは，日本語の修飾－被修飾関係の複文に対して格構造解析を行うコネクショニストモデルを追求している．本稿では，本モデルに対する計算機実験の結果を報告する．

2. ネットワーク構成

提案するモデルのネットワーク構成を図1に示す．このモデルは単文パーサ，スタック，セグメンタと呼ぶ3つのモジュールで構成される．

実行モードにおいて，全モジュールでのネットワークを形成し，1つのモジュールの出力が他のモジュールの入力になる．そして，修飾－非修飾関係にある複文を入力し，節ごとの格構造表現を出力する．

本モデルでは単語の表現に分散表現を採用している．つまり，単語を実数ベクトルで表している．以下このベクトルを**概念ベクトル**と呼ぶ．本稿では，概念ベクトルを，各要素が[0.0,1.0]の12次元の実数ベクトルとし，ランダムで固定とした．

2.1 単文パーサ

本モデルの主要なモジュールの1つが単文パーサである．単文パーサのタスクは入力文の格構造表現の出力である．本稿ではフィルモアの提唱した格文法に基づき，8つの格を設定した．名詞に対しては動作主格(AGT)，対象格(OBJ)，目標格(GOL)，場所格(LOC)，道具格(INS)を動詞に対しては動作格(ACT)，動作格過去(ACT-P)，動作格現在進行(ACT-PR)を設定した．

単語数の異なる文を取り扱うために，ネットワークには系列学習可能なElmanネットを採用した．入力層は24ユニット（12ユニット×2スロット）設定し，文節（自立語＋付属語）をつくる2つの単語に対応する2つの概念ベクトルを入力する．また，出力層は96ユニット(12ユニット×8スロット）設定した．ネットワークが学習に成功すると，実行モードにおいて入力された自立語の概念ベクトルが出力層中の正しい格スロットを埋めるようになる．出力層に現れた概念ベクトルの単語は，入力単語のうち最もユークリッド距離で近いものが同定される．

2.2 スタック

スタックは図1中に示すように，RAAM(Recursive Auto Associative Memory)ネットワークで構成される．スタックはその名のとおり，コンピュータのデータ構造であるスタックと類似の動作を実現できる．そのために，入力パターンと出力パターンを同一にする自己相関学習を行う．スタックの主な役割は，入力文の単語系列を文頭から入力し，単文ごとに文末から出力することで，単文パーサに対して逆順入力を実現することである．

2.3 セグメンタ

セグメンタは図1中に示すように，系列学習可能なJordanネットワークと自己相関ネットワークとを組み合わせたネットワークで構成される．セグメンタは全ての他のモジュールを統括する役割を担う．

セグメンタは複文を単文に切り分けるために，入力単語を監視し，動詞が入力された後に，スタックにポップ動作を単文が出力されるまで連続して行わせ，単文パーサへの情報通過ゲートを開く制御信号を出力する．

セグメンタの入力層には入力単語の概念ベクトルと，スタックの中間層の活性度パターンが入力される．出力層には，入力層への入力と同数のユニットに加え，制御信号の6ユニットが用意される．制御信号とは3つのモジュールの間の情報伝達をコントロールするためのゲート信号を意味する．出力層には入力層と同じパターンが出力されるように自己相関学習され，制御信号ユニットは，入力単語系列によって全モジュールが適切なタイミングで全ての動作が行われるような信号が時系列で出力されるように学習される．

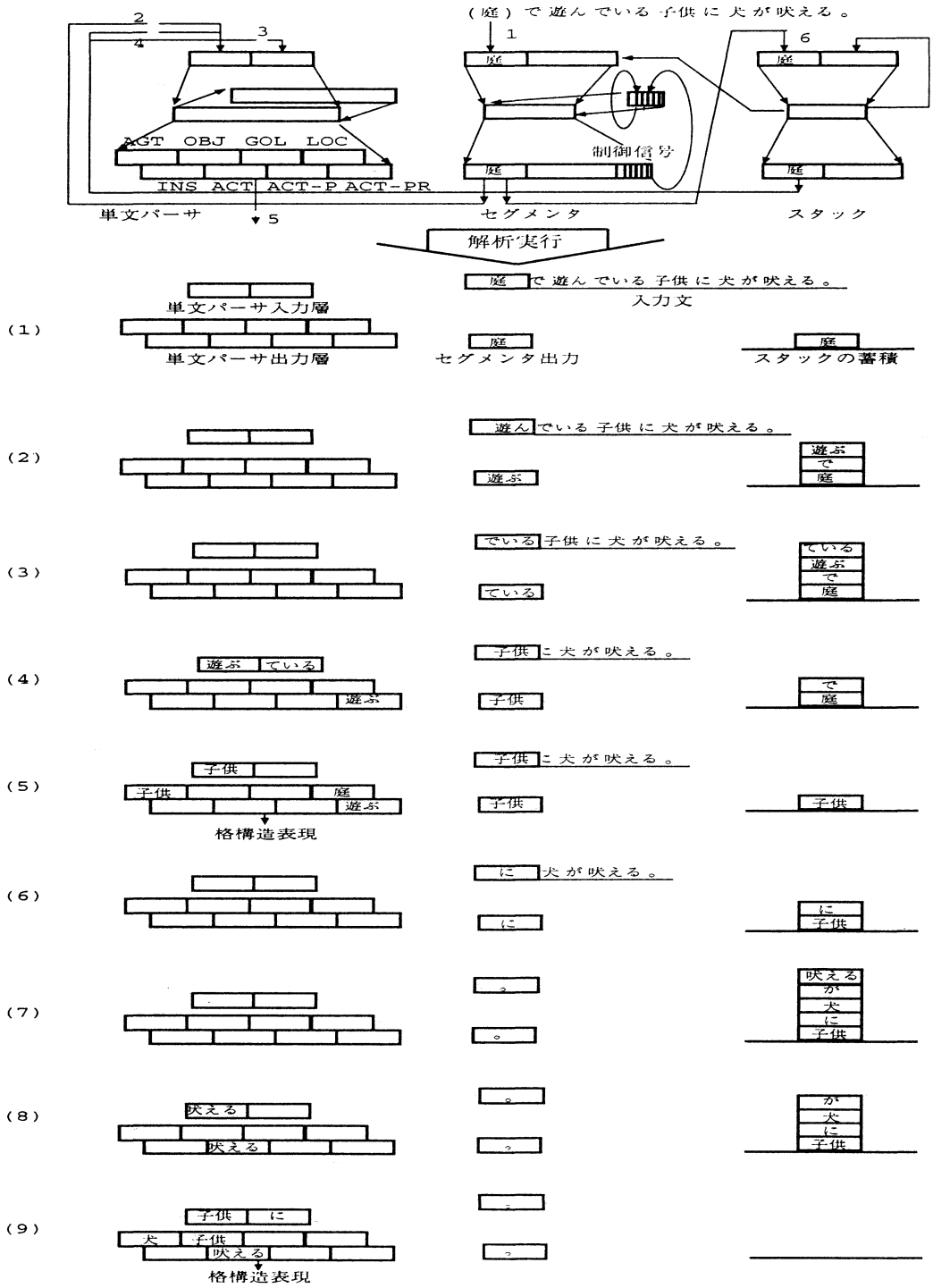

図 1: ネットワーク構成と格構造解析実行過程

3. 実行過程

今，次のような複文がシステムに入力されたとする．

- 庭で遊んでいる子供に犬が吠える。

この複文では，修飾節と主節にそれぞれ以下のような格役割が与えられること（本モデルの場合，直接には，単文パーサの適切な格スロットに適切な単語の表現が出力されること）がシステムの行うタスクとなる．

- 庭で　子供が　遊ぶている
  場所格　動作主格　動作格（現在進行）
- 子供に　犬が　吠える
  対象格　動作主格　動作格

図1に沿いながら順に説明する．

1. 「庭」がセグメンタに入力する．「庭」をセグメンタの出力層から取り出し，スタックの入力層にプッシュする（図1-(1)）．続く単語も同様にスタックにプッシュしていく．
2. 動詞「遊ぶ」をセグメンタが検出する．「遊ぶ」をセグメンタの出力層から取り出し，スタックの入力層にプッシュする（図1-(2)）．
3. 動詞「遊ぶ」の次の単語が助動詞「ている」なので，助動詞「ている」をスタックにプッシュする（図1-(3)）．
4. 修飾語「子供」がセグメンタを入力したら，一旦入力を抑止する．助動詞「ている」をスタックからポップし，単文パーサの付属語入力スロットに伝達する．そして，動詞「遊ぶ」をスタックからポップし，単文パーサの自立語入力スロットに伝達する．その後，単文パーサが，動詞「遊ぶ」を出力層の動作格（現在進行）スロットに出力する（図1-(4)）．
5. 次に，スタックにプッシュされている「で」をポップし，単文パーサの付属語入力スロットに伝達する．そして，スタックにプッシュされている「庭」をポップし，単文パーサの自立語入力スロットに伝達する．その後，単文パーサが，「庭」を出力層の場所格スロットに出力する）．次に，修飾語「子供」であるが，これはセグメンタの出力層から，単文パーサの自立語入力スロットに伝達する．また同時に，主節の解析のため，この修飾語「子供」は，セグメンタの出力層から出力し，スタックへのプッシュも行う．その後，単文パーサが，修飾語「子供」を出力層の動作主格スロットに出力する．ここで，従属節の格構造表現を単文パーサが出力する（図1-(5)）．主節の解析のため，単文パーサの文脈層をクリアする．これから，主節の解析に入る．
6. 「に」をセグメンタの出力層から取り出し，スタックの入力層にプッシュする（図1-(6)）．
7. 「続く単語「犬」,「が」,「吠える」も同様にスタックにプッシュしてく．次に句点「。」をセグメンタに入力する（図1-(7)）．
8. 句点「。」をセグメンタが検出すると，動詞「吠える」をスタックからポップし，単文パーサの自立語入力スロットに伝達する．その後，単文パーサが，動詞「吠える」を出力層の動作格スロットに出力する（図1-(8)）．
9. 次に，単語「が」,「犬」をスタックからポップし，単文パーサの入力層に伝達する．単文パーサが，単語「犬」を出力層の動作主格スロットに出力する．次に，単語「に」,「子供」をスタックからポップし，単文パーサの入力層に伝達する．単文パーサが，単語「子供」を出力層の対象格スロットに出力する．そして，主節の格構造表現を単文パーサが出力する（図1-(9)）．最後に，単文パーサの文脈層をクリアする．

表1: 使用単語

名詞 (12)	妻 子供 犬 猫 雪 雨 雪割草 木 塀 庭 花壇 犬小屋
動詞 (8)	登る 入る 走る 遊ぶ 降る 植える 吠える 叱る
助詞 (4)	が を で に
助動詞 (2)	た ている
その他 (2)	。 NULL

4. 実験

4.1 データ

計算機実験に用いた単語は表1に示す計28単語である．これらの単語を使用し，学習，テストのための複文データを計算機を用いて自動生成した．生成した文に対し，日本人である筆者の言語感覚で日本語として正しいものだけ選別し，さらにそれらに助動詞を付加してより自然になるように作成した．作成した文は，単文50文，修飾節の深さが1の複文503文，修飾節の深さが2の複文399文の計952文であった．今回の実験では，主節が現在時制の文のみしか作成しておらず，過去時制，現在進行時制の文は作成しなかった．

4.2 実験条件

学習に用いたデータは修飾節の深さ1の複文503文のうちの70%である353文である．単文と修飾節の深さ2の複文は学習には用いず，解析テストに用いた．

学習方式は，全てのモジュールを1つとして，同時に同期をとりながら学習する方式と，各モジュール個別に学習する方式とが考えられる．さらに詳細に分け

ると，いくつかの学習方式が考えられるが，本実験では，全てのモジュールを1つとして学習するが，制御信号のみ教師信号を用いて復元する方式を採用した．

各モジュールのネットワーク条件は以下のとおりである．全モジュールとも中間層数は1層とした．単文パーサの中間層ユニット数，文脈層ユニット数は90とした．セグメンタの中間層ユニット数は60とした．スタックの中間層ユニット数は60とした．すなわち，セグメンタの入力層は概念ベクトルの12ユニットとあわせて72ユニットで構成される．また，スタックの入力層，出力層も概念ベクトルの12ユニットとあわせて72ユニットで構成されることになる．全モジュールとも，シグモイドの傾きは1.0とした．

学習アルゴリズムは純粋な誤差逆伝搬法 [3] を用いた．学習方法は，逐次学習法を用い，モジュールの実行動作が前方向に1回行われるたびごとに，誤差を逆方向に1回伝搬した．学習条件は以下のとおりである．全モジュールとも学習係数 η は0.1で慣性項はなしとした．セグメンタの残存率 γ は0.5とした．結合荷重と各ユニットのバイアスの初期値は $[-1.0, +1.0]$ でランダムに設定した．

学習の終了条件は，単文パーサにの出力層において1ユニットあたりの平均RMS(Root Mean Square)誤差が0.015より小さくなった場合とした．この時の学習回数は654回であり，セグメンタとスタックの出力層における1ユニットあたりの平均RMS誤差はそれぞれ0.0263，0.0164であった．

学習終了後のネットワークの動作条件は以下のようにした．単文パーサの出力層での単語の同定はユークリッド距離を用い，入力単語の中で最も近い距離の単語を出力単語とした．セグメンタの出力層におけるコントロールユニットが0.5より大きければ1の制御信号，0.5以下であれば0の制御信号とみなした．また，解析テストでは，単文パーサからの格構造表現が，全て正しいタイミングで，全ての格スロットに対して正しい単語が同定された場合を正解とした．

4.3 結果

学習終了後，単文，修飾節の深さ1の複文，修飾節の深さ2の複文に対して，解析テストを行った．この結果を図2に示す．未学習の修飾節の深さ1の複文と未学習の修飾節の深さ2の複文とが，それぞれ89.3%，79.2%という正解率で格解析を行なえている．

5. 考察

実験結果より，本モデルは，修飾節の深さが1の文のみ学習させ，修飾節の深さが2の文に対しても正しく格構造解析が行えるという構造上の一般化を，79.2%の正解率で実現できたと言える．

修飾節の深さが2の文のうちで間違って解析された文(83文)の出力結果を追ってみた．すると，セグメンタの制御信号が間違っているものは35文，制御信号は正しいが，単文パーサの出力が間違っているものは48文あった．これらは，セグメンタの中間層ユニット数，単文パーサの中間層ユニット数をそれぞれ増やすことで，RMS誤差を今回の実験より小さい値まで学習を進ませることができ，その結果正解率は向上すると考えられる．

表 2: 文構造別解析結果

文構造	正解	/	文数	正解率 (%)
修飾節の深さ1 (学習文)	330	/	353	93.5
単文	29	/	50	58.0
修飾節の深さ1 (未学習文)	134	/	150	89.3
修飾節の深さ2	316	/	399	79.2

参考文献

[1] Miikkulainen,R.: " Subsymbolic Parsing of Embedded Structure," *Computational Architecutures Integrating Neural and Symbolic Processes, edited by R.Sun and L.A.Bookman*, pp.153–186 (1996).

[2] Ajay N. Jain: " Generalization Performance in PARSEC-A Structured Connectionist Parsing Architecture," NIPS4, KAUF, edited by John E. Moody and Stephen J. Hanson and Richard P. Lippmann, pp.209–219 (1992).

[3] Rumellhart,D.E., Hinton, G.E., and Williams, R.J.: *Learning Internal Representations by Error Propagation*, in Rumelhart.D.E., McClelland,J.L., and the PDP Research Group. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition* Volume 1, pp. 318–362, The MIT Press, Massachusetts. (1986).

[4] 本木実, 渡辺啓, 嶋津好生: 日本語複文のためのコネクショニストパーサ, 情報処理学会九州支部研究会報告, pp. 168–176 (1998).

付録

以下に本研究に用いた複文の例を示す．

- 子供が塀に登る。
- 塀に子供が登る。
- 雪が庭に降る。
- 雪が降る庭を猫が走る。
- 猫が雪が降る庭を走る。
- 犬が走るている庭に降る雪で子供が遊ぶ。
- 雪が降るている庭を走る犬を妻が叱る。

単語の表現には，形態素解析処理後の単語を想定しているため「走っている」は「走る ている」となる．