コネクショニストモデルを用いた日本語単文解析モジュール

本木 実・渡邊 啓・篠原 理一郎・嶋津 好生
九州産業大学工学部

1.はじめに

我々の研究室ではコネクショニストモデルを用いた日本語理解システムの研究を過去5年に渡って行ってきた[1]. コネクショニストモデルとは人工ニューラルネットワークモデルであり, これまで形態素抽出モジュール, 文生成モジュール等を開発してきた. 現在, 日本語複文解析モジュールの概念設計およびシミュレーション実験中である. 今回はこの複文解析モジュール中に使用される単文の文解析モジュール (単文パーサ) について述べる.

2.日本語単文解析モジュール

2.1 格構造解析

文解析モジュールは文の構成要素に対する格の認定をタスクとする. 今回我々は Fillmore の提唱した格[2]をもとに表1に示す7つの格を設定した.

表1 設定した格

主題格	その文の主題となる者の役割
主格	ある動作を引き起こす者の役割
対象格	移動する対象物や変化する対象物. あるいは, 判断, 想像のような心理事象の内容を表わす役割
道具格	ある出来事の直接原因となったり, ある心理事象と関係して反応を起こさせる刺激(stimulus)となる役割
位格	ある出来事が起こる場所および位置を表わす役割
動作格	述語動詞を表わす役割
動作格過去	述語動詞の過去時制を表わす役割

主題格については, 我々の研究室で考案したものである. 何故なら, 例えば「パンは少年が公園で食べる。」などの文で, 主格を導く格助詞「は」「が」がそれぞれ「パン」, 「少年」に付属しているが, 主格は「少年」であり, 「パン」は対象格になる. 文のテーマとしては「パン」を主題として扱っており, これを捉えるために「主題格」の存在が必要と考えたからである.

2.2 ネットワークアーキテクチャ

Miikkulainen らの英文パーサ[3]と同様, 日本語の文解析モジュールは系列学習の可能な3層階層型のElman ネットワークを採用した (図1). 入力層(24ユニット) には, 自立語と付属語の2つのアセンブリを用意し, 各単語に対応させた各要素が[0.0,1.0]の12次元の概念ベクトル (3.1.2 参照)をそれぞれ入力する. 出力層は7アセンブリから成り7つの格スロット与え, 計84ユニット用意する.

単文パーサのタスクは, 入力文中の自立語への格の割り付けである. 入力文の単語に対応する概念ベクトルを, 自立語と付属語のペアで文節単位に入力し, 自立語の概念ベクトルが適切な格スロットへ落ち込むように, 逆誤差伝播学習アルゴリズムを用いて学習させる. 出力アセンブリに現れたベクトルにユークリッド距離が最も近い単語を出力単語だと識別する.

3.実験

3.1 実験データ

3.1.1 使用単語および文章

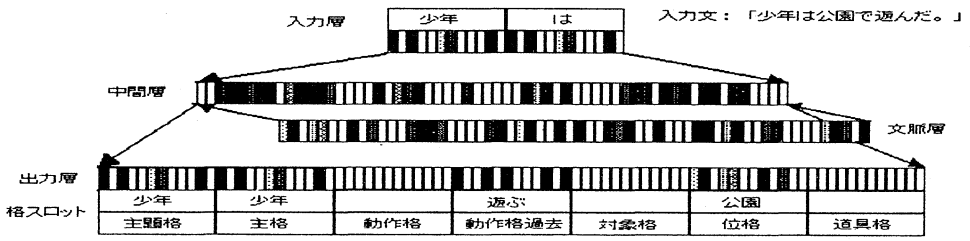


図1 日本語単文解析モジュール

実験に使用した文章は，表2に示す名詞，動詞，助詞，助動詞を含む42単語を用いて，図2のテンプレートをもとに手作りで作成し，現実世界で意味の通る文のみ抜き出して計3812文作成した．

表2　使用単語

名詞 (19個)	少年,少女,犬,馬,イルカ,向日葵,松,ちょうちょ,あり,庭,草原,海,ボール,ラケット,ズボン,スカート,本,みかん,餌
動詞 (15個)	飛ぶ,走る,咲く,遊ぶ,泳ぐ,見る,着る,植える,読む,買う,食べる,打つ,投げる,運ぶ,乗る
助詞 (5個)	は,が,を,で,に
助動詞 (2個)	た,だ
その他 (1個)	。

テンプレート1　補足成分+統括成分。
テンプレート2　補足成分+補足成分+統括成分。
テンプレート3　補足成分+補足成分+補足成分+統括成分。

＜補足成分＞＝＜名詞＞＜格助詞＞
＜統括成文＞＝＜動詞＞｜＜動詞＞＜助動詞＞

図2　文章作成テンプレート

3.1.2　概念ベクトル

各単語は，各要素が[0.0,1.0]の12次元の概念ベクトルと呼ばれるベクトルで表現される．この概念ベクトルはランダムに与えた固定表現と，Miikkulainen のFGREP方式で獲得した適応表現の2種類が考えられる．FGREP方式とは拡張逆誤差伝播アルゴリズムを言い，格構造解析のタスクを学習しながら，同時に各単語が適応的に概念ベクトルを獲得する方式を言う．今回の報告には固定表現での結果を示す．

3.2 学習方式

学習方式として次の3種類が考えられる．例えば，入力文「少年は公園で遊ぶ。」での教師信号の提示の仕方として図3に示す3種類について実験を行う．また，日本語は主要語が後ろに来ることから，入力する語順を入れ替え，文末から入力した方が良いと考えられる．このことを確かめるために，文頭から入力する正順入力と，文末から入力する逆順入力についてそれぞれ調べる．

主題格	主格	動作格	動作格過去	対象格	位格	道具格
少年	少年					
					公園	
		遊ぶ				

(a)ゲート作用学習

主題格	主格	動作格	動作格過去	対象格	位格	道具格
少年	少年					
少年	少年				公園	
少年	少年	遊ぶ			公園	

(b) スロット保持学習

主題格	主格	動作格	動作格過去	対象格	位格	道具格
少年	少年	遊ぶ			公園	
少年	少年	遊ぶ			公園	
少年	少年	遊ぶ			公園	

(c)完全予測学習

図3　学習方法（正順入力の場合）
1段目，「少年は」を入力．2段目，「公園で」を入力，3段目：「遊ぶ。」を入力

3.3 実験内容

3.3.1 基礎実験

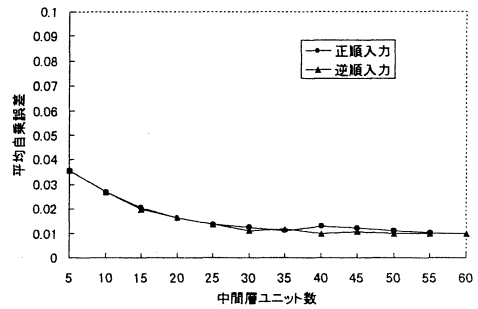
3.1.1 で説明した文章のうち各テンプレートからそれぞれ30%をランダムに選別した1143文を学習文として，3種類の学習方式において，正順入力，逆順入力の計6種類のシステムで学習を行った．学習は1ユニットあたりの平均自乗誤差が 0.01 以下となり収束するか，または学習回数が200回になるまで行った．中間層のユニット数を次第に増やしながら実験を行った．シグモイドの傾きは0.6とし，学習パラメータは，学習係数η=0.4，モーメント係数α=0.4である．

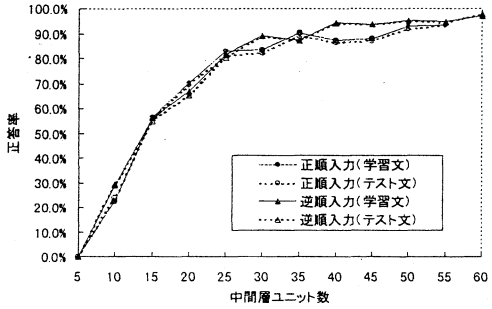
学習が収束した後，学習文に対して格割り付けのタスクの正答率を調べた．正答率は，文節が入力される全てのタイミングにおいて，全格スロットが正しく出力されることをその文章の正答として，全文章に対する百分率で表した．

また，汎化能力を調べるために，学習実験に用いた残りの70%の2669文をテスト文として格割り付けのタスクを行わせ，正答率を調べた．

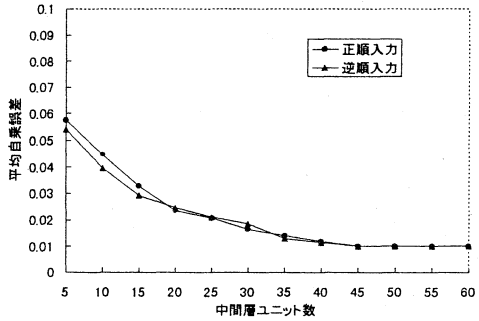
3.3.2 正答率の追求

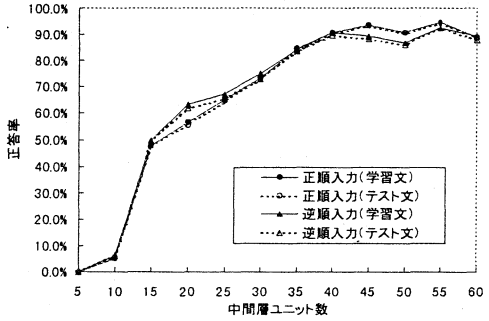
次に，更に学習を進めるとコネクショニストモデルが単文パーサのとしてどれだけの正答率を有することが可能であるかを調べるために学習を行った．中間層のユニット数を 90 としスロット保持学習を 1000 回まで行い，正答率を調べた．



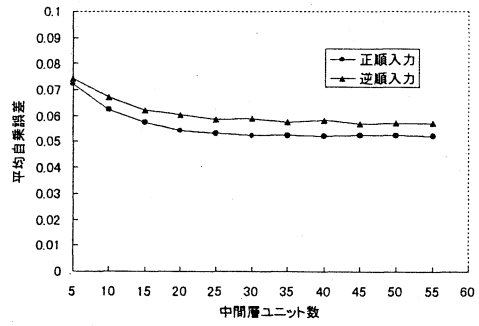


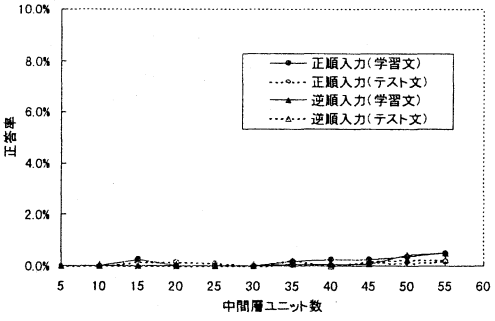
(a)ゲート作用学習





(b)スロット保持学習





(c)完全予測学習

図4　中間層のユニット数に対する平均自乗誤差と正答率(学習回数２００回時)
（シグモイドの傾き=0.6,　学習係数η=0.4,　モーメント係数α=0.4）

4.実験結果

4.1 基礎実験

図4に各学習方式での実験結果を示す.

完全予測学習については正順入力, 逆順入力のどちらも学習が収束せず, その正答率はかなり低かった. また, ゲート作用学習, スロット保持学習において, 正順入力と逆順入力の結果を比較してみると, 学習の収束は両者に大きな違いは見られず, この条件での学習は正順入力と逆順入力の差が無いことがわかる. 中間層ユニット数は50以上必要であることがわかった.

4.2 正答率の追求

中間層のユニット数を90, 学習を1000回まで行った場合の結果を表3に示す. 正順入力, 逆順入力ともに汎化能力約 99%以上の結果が出ていることがわかる.

表3 学習回数1000回時の正答率

	平均自乗誤差	正答率(テスト文)
正順入力	0.002378	99.78 %
逆順入力	0.002648	99.81 %

5.検討

日本語の主要部は, 名詞+格助詞では格助詞, 補足成分と統括成分では統括成分というように, 構成要素の後方に位置する. 更に, 我々日本人も文末の動詞が来て初めて文の構成要素がそれぞれどのような意味合いを持つのかがはっきりする. また, Elman ネットワークは初めに入力された情報が後に入力された情報に加味して処理され, その情報は初めであればあるほどその後の処理に影響を及ぼす. これらのことから, 文末から先に入力する逆順入力の方が, 格解析のタスクを良くこなせると予想されたが, そのような結果は明瞭には得られなかった. しかし, 学習文とテスト文の正答率が殆ど差がないこと, それぞれのテンプレートから等しく学習文とテスト文に30%, 70%の割合で分けていることから, 学習文とテスト分の選び方のバランスが良すぎたのではないかと思われる.

今後, これらの割合を変えてみて, 例えば10%を学習文, 残り90%をテスト文というように, 多少負荷を増やして実験を行ってみる必要がある. このことは, 今後単語数を増やしていったときに, どれだけの汎化能力を有することが可能であるかという点でも重要である.

次に, 正答率が99%以上と良い結果が出ている. 誤答したデータを調べてみると, スロット位置は正しいが他の単語と誤認識した例が存在した. これはFGREP方式で獲得した概念ベクトルを用いた場合に意味のあるものになると思われる. 何故なら, 今回報告しなかったが, FGREP方式で獲得した概念ベクトルは, 学習データを反映することができ, 各品詞ごとに, また意味の類似した単語ごとにベクトルの距離が近くなるからである.

6.おわりに

コネクショニストモデルを用いた単文解析モジュールのアーキテクチャと, その学習実験を行った結果について述べた. 正答率は高かったものの, 格スロットの数が少なく, そのため学習文が単純すぎたためであると思われる. 今後単語数を増やし, 単文だけでなく複文の解析を行えるようにするためにも, 学習文の選定を吟味する必要がある.

また, 現実世界に意味の通らない文, 例えば「海が走る。」などの文を入力した場合, どのような結果が出力がされるかも確かめなくてはならない.

参考文献

[1]原田信義, 山村邦彦, 嶋津好生 : コネクショニストモデルによる日本文の格構造解析, 情報処理学会九州支部研究会報告, pp.251-260,March 1996.

[2] C.J.Fillmore, 田中春美・船城道雄訳 : 格文法の原理, pp283, 三省堂, 1975.

[3]R.Miikkulainen : Subsymbolic Parsing of Embedded Structures , in Computational Architectures Integrating Neural and Symbolic Processes, edited by R. Sun and L. A. Bookman, pp153-186, kluwer Academic Publishers, 1996.

[4]渡邊啓 : コネクショニスト日本語処理システムの構成的研究, 九州産業大学大学院修士論文,平成 9 年度.

[5]中野馨 : ニューロコンピュータの基礎, コロナ社, 1991.