

コネクショニストの自然言語処理

嶋津 好生

九州産業大学工学部

日本語複文の単語系列を深層格構造に変換するシステムをいくつかの階層型人工神経回路網を結合したマルチネットシステムで実現する。このシステムは日本語の再帰的構造や構成要素が処理できる。かかるコネクショニスト自然言語処理システムにおいては言語そのものの構造に関して再考を要し、また人工神経回路網上に言語現象をマッピングする方法に関して多くの作業仮説を必要とする。

Connectionist Natural Language Processing

Yoshio SHIMAZU

Faculty of Engineering, Kyushu Sangyo University

We realize the multi-net system that output a set of deep case structures, being input a sequence of words of a Japanese sentence with embedded clauses, and so is able to process the recursive and constituent structures of the language. Such a connectionist natural language processing system make us reconsider the structure of the language itself and need many working hypotheses concerning how to map the language phenomena onto the artificial neural networks.

言語獲得理論 チョムスキーは、文法には経験のみから帰納出来ない部分が含まれており、また、他の知識を学習する際に働くと思われる一般的な心的機構では文法の獲得を説明できないとして、文法は生まれつき頭の中に備わっているとした。ここでいう「文法」とは日本語や英語のような個別言語の文法ではなく、あらゆる言語に普遍的に備わっている「普遍文法」である。チョムスキーの言語獲得理論は「原理とパラメータの理論」と呼ばれる。大人の成熟した文法は生得的に与えられた普遍文法と経験との関数である。普遍文法はパラメータを含む一般原理から成る体系であり、生後環境から得られる経験に矛盾しないように個々の原理のパラメータの値が固定されていき、その結果特定の個別言語が獲得されるのだという。たとえば、動詞句は統括成分である動詞に補足成分である幾つかの格要素の名詞が付属している。普遍文法ではこの動詞句の中の語順は指定しない。語順はパラメータとして含まれる。このパラメータの値は二者択一で、日本語と英語とでは、経験を通して違う値に固定される。普遍文法を追求する言語学的努力は、われわれヒトにとってことばの普遍性とは何かを知るため辿るべき一つの道筋であることはたしかである。

生得論といわれる普遍文法の理論においても、パラメータ値の設定は経験によるとしている。言語獲得の発達過程は普遍文法のような内的制約と経験による外的制約の複雑な相互作用に基づくとの認識になってきた。そして言語獲得に関する最近の論争は、生得性が普遍文法のような表象なのか、生物学的器質的機構なのかに移行している。後者に属するパラダイムの一つにコネクショニズムがある。

コネクショニズム 脳の神経回路網のモデルを人工神経回路網とかコネクショニストモデルとか呼んでいる。コネクショニズムは、人工神経回路網を道具として使い、一般的な心理学的発達経路の背後にある複数の内的および外的制約の複雑な相互作用を解明するためのパラダイムである。勿論、言語獲得の研究にも貢献できる。ただしコネクショニズムそれ自体は発達の理論ではなく、むしろ、発達に関する仮説が与えられたのを受け、そのモデルを構築して検証するための道具である。コネクシ

ョニストモデルを構成する要素は、神経細胞ユニットとネットワークアーキテクチャ、それに学習アルゴリズムと学習データである。ユニットとネットワークアーキテクチャは内的制約、学習データは外的制約であるが、学習アルゴリズムはパラメータ値の選択を含む内的制約であって外的制約でもある。このようにコネクショニストモデルは生得論的でもあり経験論的でもある。両方の要素を調節することによって、試行錯誤的に柔軟にモデルの改良をはかることができる。

チョムスキーの普遍文法はメタ言語であり知識である。この場合の「知識」は言語行動を支える表象を指している。さてコネクショニズムにおいては「表象」ということばの定義を人工神経回路網で操作的に実現可能なものとして考えなければならない。すなわち表象とは人工神経回路網の活性パターンであり、それはさらに神経細胞ユニット間の重み付けられた結合のパターンに依存する。結合の重み付けは可塑性シナプス結合のモデルであり、ユニットの活性化を通して学習される。はじめから固定的な値を持って発生するものではない。したがってコネクショニズムにおいては表象に関するかぎり生得的であり得ない。チョムスキーの「原理とパラメータの理論」が表象レベルの生得主義であるならば、これを否定せざるを得ない。

ことばの表象について考えよう。文は形態素の順列である。形態素は品詞に分類される。形容詞、名詞、副詞、動詞、形容動詞などの品詞は概念を担う品詞ということで、概念語と呼ぶ。形容詞や形容動詞は二つの働き、装定的機能と述定的機能を持つ。助詞、助動詞、接続詞、代名詞、関係代名詞などは概念語と区別して機能語と呼ぶ。形態素の表象はいくつかの神経細胞ユニットの集まりの活性パターンである。これを、アセンブリパターンと呼ぼう。学習が収束し表象が固定化することで語彙の獲得が完了する。文の理解の過程で形態素の表象はネットワークの上を転移していく。文の表象が形成されることで理解が完了する。概念語の表象は理解の完了時まで維持されるが、機能語の表象は概念語表象の転移を支援して役目が終われば消滅する。複文や重文を含めて「文」というならば、代名詞を除いてすべての機能語の働きがその文の理解過程内に閉じている。代名詞だけが先行する文の記憶を必要とすることもある。

動詞や述定形容詞・形容動詞を統括成分として備えているが文の一部となっているものを「節」と呼ぶ。文中で一つの品詞に相当する働きをする語群で、その中に統括成分を含まぬものを「句」と呼ぶ。文の理解過程には次のような操作が実行される。

1) 句の表象は単一の品詞並に圧縮される。また、装定形容詞+名詞、副詞+動詞、副詞+述定形容詞・形容動詞などの表象も単一の名詞や動詞並に圧縮される。そして動詞に付属している助動詞、終助詞、接続助詞などの表象も単一の品詞並に圧縮される。

2) 文は節に分割される。

3) 各節はその格構造の表象に変換される。同時に各格構造間の関係も表象されなければならない。文が修飾-被修飾関係の複文である場合は、格構造間で同一の名詞が含まれていれば、そのことが表象される。また文が重文である場合は、格構造間の時間関係や因果関係が表象される。以上のすべてが揃えば、それが文の表象である。

格構造の表象について考えよう。コネクショニストモデルにおいて出現する「表象」は、ネットワークの「どこがどのように活性化しているか」である。「どこが」とはユニットアセンブリの位置のことであり、「どのように」とはアセンブリパターンのことである。格構造の表象の場合、異なる位置のユニットアセンブリを異なる格の名詞アセンブリパターンが入る場所とする。これを格スロットと呼ぶ。同様に動詞についても、その異なるテンス、アスペクト、ヴォイス、モダリティに対して異なるスロットを用意する。統括成分である動詞のアセンブリパターンと補足成分であるいくつかの名

詞のアセンブリパターンとがそれぞれ同時に適当なスロットを満たしたとき格構造の表象が出現する。

また、格構造間で同一の名詞であることを表象するには、名詞のユニットアセンブリにいくつかユニットを追加して、同一名詞のその部分が呼応して同一のパターンで活性化するようにすればよい。格構造間の時間関係や因果関係を表象するには、動詞のユニットアセンブリに同様なことを行い、ただ違うところは、関係の方向性を加味して呼応する活性パターンを用意することである。

実際に日本語の文解析モジュールを実現してみた。ネットワークアーキテクチャは、いくつかの階層型ネットワークをさらにネットワーク結合したものである。そして、必要なところにシンプルリカレントネットワークを組み込んでいる。学習アルゴリズムは誤差逆伝搬学習則だけで十分であった。

コネクションリスト複文パーサ 図1に具体的な日本語の複文によって作られる深層格構造の集まりを示す。図2に日本語の複文パーサを示す。複文パーサは5つの階層型ネットで構成される。左からそれぞれ単文パーサ、セグメンタ、スタック、コンプレッサ、ワンダラーコンプレッサである。それぞれ学習が収束し期待された機能をはたすものとする。セグメンタの入力層に複文の単語系列が入力され、単文パーサの出力層にいくつかの深層格構造がつつぎつつぎに出力される。

実際に日本語複文「私は無邪気に子供達が遊ぶ賑やかな公園で不意に女を殴る男を見た。」を入力して複文パーサの働きを辿って見よう。

- 1) 「私」が入力される。
 1. 1) 「私」はセグメンタを経由してスタックにプッシュされる。
 1. 2) 次単語入力 of 準備をする。(セグメンタの状態層がクリアされる.)
- 2) 「に」が入力される。
 2. 1) 「に」はセグメンタを経由してスタックにプッシュされる。
 2. 2) 次単語入力 of 準備をする。(セグメンタの状態層がクリアされる.)
- 3) 「無邪気に」が入力される。
 3. 1) 「無邪気に」はセグメンタを経由してワンダラーコンプレッサにプッシュされる。
 3. 2) 次単語入力 of 準備をする。
- 4) 「子供達」が入力される。
 4. 1) 「子供達」はセグメンタを経由してスタックにプッシュされる。
 4. 2) 次単語入力 of 準備をする。
- 5) 「が」が入力される。
 5. 1) 「が」はセグメンタを経由してスタックにプッシュされる。
 5. 2) 次単語入力 of 準備をする。
- 6) 「遊」が入力される。
 6. 1) 「遊」はセグメンタを経由してワンダラーコンプレッサにプッシュされる。
 6. 2) ワンダラーコンプレッサ中間層から「無邪気に遊」が取り出されスタックにプッシュされる。(ワンダラーコンプレッサ中間層がクリアされる.)
 6. 3) 次単語入力 of 準備をする。(セグメンタの状態層がクリアされる.)
- 7) 「ぶ」が入力される。
 7. 1) 「ぶ」はセグメンタを経由してスタックにプッシュされる。
 7. 2) 次単語入力 of 準備をする。(セグメンタの状態層がクリアされる.)

(複文)

私は無邪気に子供達が遊ぶ賑やかな公園で不意に女を殴る男を見た。

(単文)

無邪気に子供達が賑やかな公園で遊ぶ。

賑やかな公園で不意に男が女を殴る。

私は男を見た。

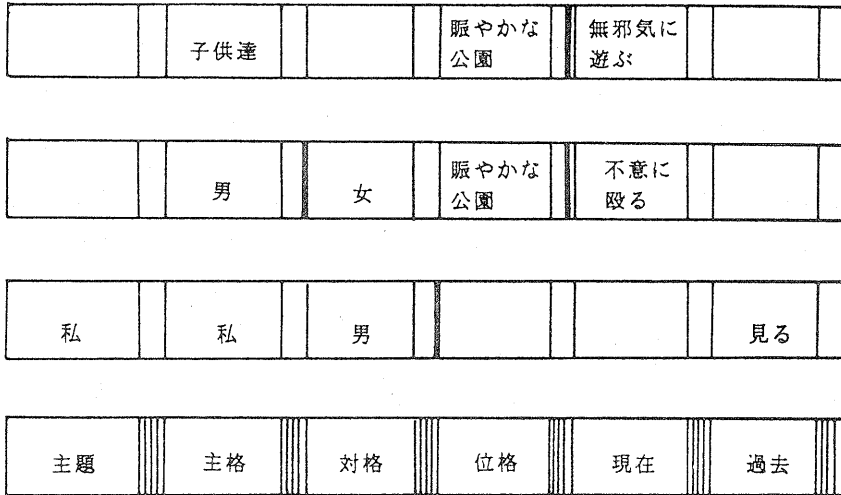


図1. 格構造パイル

格構造パイルは複文が解析されて出来た深層格構造の集まりである。名詞はケースを選び、動詞はテンスを選んで転移する。

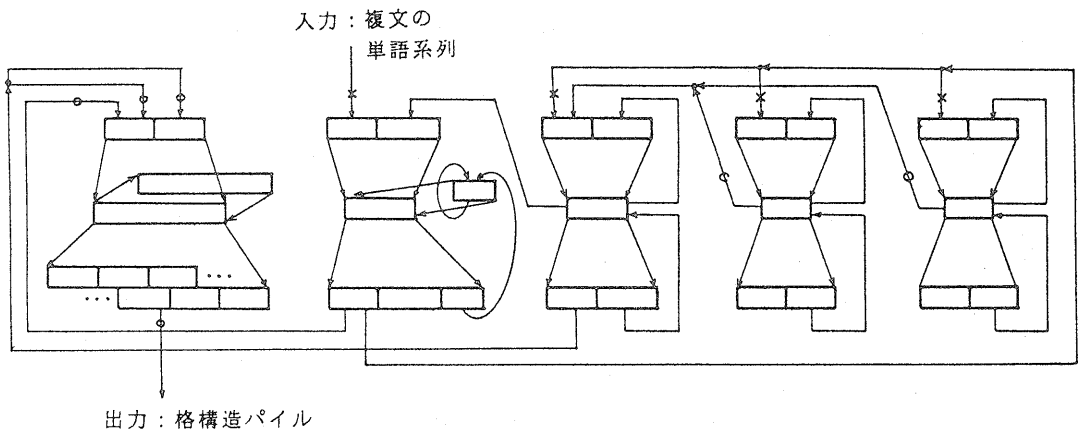


図2. 複文パーサ

複文パーサは5つの階層型ネットで構成される。左からそれぞれ単文パーサ、セグメンタ、スタック、コンプレッサ、ワンダラーコンプレッサである。セグメンタの入力層に複文の単語系列が入力され、単文パーサの出力層から格構造パイルが出力される。

- 8) 「賑やかな」が入力される。
8. 1) 「賑やかな」はセグメントを經由してコンプレッサにプッシュされる。
8. 2) 次単語入力 of 準備をする。
- 9) 「公園」が入力される。
9. 1) 「公園」はセグメントを經由してコンプレッサにプッシュされる。
9. 2) スタックから「ぶ」がポップされ単文パーサ入力層付属語アセンブリに入力される。
9. 3) スタックから「無邪気に遊」がポップされ単文パーサ入力層自立語アセンブリに入力される。(単文パーサ出力層動詞現在スロットに「無邪気に遊」が出力されている。)
9. 4) スタックから「が」がポップされ単文パーサ入力層付属語アセンブリに入力される。
9. 5) スタックから「子供達」がポップされ単文パーサ入力層自立語アセンブリに入力される。(単文パーサ出力層動詞現在スロットに「無邪気に遊」が、名詞主格スロットに「子供達」がそれぞれ出力されている。)
9. 6) コンプレッサ中間層から「賑やかな公園」が取り出されスタックにプッシュされる。
9. 7) スタックから「賑やかな公園」がポップされ単文パーサ自立語アセンブリに入力される。(単文パーサ出力層動詞現在スロットに「無邪気に遊」が、名詞主格スロットに「子供達」が、名詞位格スロットに「賑やかな公園」がそれぞれ出力されている。)
9. 8) コンプレッサ中間層から「賑やかな公園」が取り出されスタックにプッシュされる。(コンプレッサ中間層がクリアされる。)
- 9. 9) 単文パーサ出力層から格構造出力が取り出される。(単文パーサ文脈層がクリアされる。)
9. 10) 次単語入力 of 準備をする。(セグメントの状態層がクリアされる。)
- 10) 「で」が入力される。
10. 1) 「で」はセグメントを經由してスタックにプッシュされる。
10. 2) 次単語入力 of 準備をする。(セグメントの状態層がクリアされる。)
- 11) 「不意に」が入力される。
11. 1) 「不意に」はセグメントを經由してワンダラーコンプレッサにプッシュされる。
11. 2) 次単語入力 of 準備をする。
- 12) 「女」が入力される。
12. 1) 「女」はセグメントを經由してスタックにプッシュされる。
12. 2) 次単語入力 of 準備をする。
- 13) 「を」が入力される。
13. 1) 「を」はセグメントを經由してスタックにプッシュされる。
13. 2) 次単語入力 of 準備をする。
- 14) 「殴」が入力される。
14. 1) 「殴」はセグメントを經由してワンダラーコンプレッサにプッシュされる。
14. 2) ワンダラーコンプレッサ中間層から「不意に殴」が取り出されスタックにプッシュされる。(ワンダラーコンプレッサ中間層がクリアされる。)
14. 3) 次単語入力 of 準備をする。(セグメント状態層がクリアされる。)
- 15) 「る」が入力される。
15. 1) 「る」はセグメントを經由してスタックにプッシュされる。
15. 2) 次単語入力 of 準備をする。(セグメント状態層がクリアされる。)

- 16) 「男」が入力される。
16. 1) スタックから「る」がポップされ単文パーサ入力層付属語アセンブリに入力される。
16. 2) スタックから「不意に殴」がポップされ単文パーサ入力層自立語アセンブリに入力される。
(単文パーサ出力層動詞現在スロットに「不意に殴」が出力されている。)
16. 3) スタックから「を」がポップされ単文パーサ入力層付属語アセンブリに入力される。
16. 4) スタックから「女」がポップされ単文パーサ入力層自立語アセンブリに入力される。
(単文パーサ出力層動詞現在スロットに「不意に殴」が、名詞対格スロットに「女」が出力されている。)
16. 5) スタックから「で」がポップされ単文パーサ入力層付属語アセンブリに入力される。
16. 6) スタックから「賑やかな公園」がポップされ単文パーサ入力層自立語アセンブリに入力される。
(単文パーサ出力層動詞現在スロットに「不意に殴」が、名詞対格スロットに「女」が、名詞位格スロットに「賑やかな公園」が出力されている。)
16. 7) セグメンタの出力層から「男」を取り出して、単文パーサ入力層自立語アセンブリに入力し、スタックにプッシュする。
(単文パーサ出力層動詞現在スロットに「不意に殴」が、名詞対格スロットに「女」が、名詞位格スロットに「賑やかな公園」が、名詞主格スロットに「男」が出力されている。)
- 16. 8) 単文パーサ出力層から格構造出力が取り出される。(単文パーサ文脈層がクリアされる。)
16. 9) 次単語入力の準備をする。(セグメンタ状態層がクリアされる。)
- 17) 「を」が入力される。
17. 1) 「を」はセグメンタを經由してスタックにプッシュされる。
17. 2) 次単語入力の準備をする。(セグメンタ状態層がクリアされる。)
- 18) 「見」が入力される。
18. 1) 「見」はセグメンタを經由してスタックにプッシュされる。
18. 2) 次単語入力の準備をする。(セグメンタ状態層がクリアされる。)
- 19) 「た」が入力される。
19. 1) 「た」はセグメンタを經由してスタックにプッシュされる。
19. 2) 次単語入力の準備をする。(セグメンタ状態層がクリアされる。)
- 20) 「。」が入力される。
20. 1) スタックから「た」がポップされ単文パーサ付属語アセンブリに入力される。
20. 2) スタックから「見」がポップされ単文パーサ自立語アセンブリに入力される。
(単文パーサ出力層動詞過去スロットに「見」が出力されている。)
20. 3) スタックから「を」がポップされ単文パーサ付属語アセンブリに入力される。
20. 4) スタックから「男」がポップされ単文パーサ自立語アセンブリに入力される。
(単文パーサ出力層動詞過去スロットに「見」が、名詞対格スロットに「男」が出力されている。)
20. 5) スタックから「は」がポップされ単文パーサ付属語アセンブリに入力される。
20. 6) スタックから「私」がポップされ単文パーサ自立語アセンブリに入力される。
(単文パーサ出力層動詞過去スロットに「見」が、名詞対格スロットに「男」が、名詞主題スロットと名詞主格スロットに「私」が出力されている。)
- 20. 7) 単文パーサ出力層から格構造出力を取り出す。(単文パーサ文脈層がクリアされる。)
20. 8) 次単語入力の準備をする。(セグメンタ状態層がクリアされる。)