

Connectionist Model による構文解析モデル†

森 辰 則†† 中 川 裕 志††

本稿においてわれわれは、文法などで静的に記述しきれない言語の動的運用面を解析するためにネットワークによる超並列計算モデルである Connectionist Model による構文解析モデルを構築した。本構文解析機では計算ユニットを文法範疇に割り当て、ユニットの内部状態である活性化レベルを句構造に対する注目度の高さと見なす。さらに、語が入力される度に句構造が動的に形成されていく過程を示すために動的結合機構を導入した。これにより、動的に形成された構造の結合の強さを活性化レベルにより示すことができ、より詳細な議論ができる。また、句構造規則を活性化の順序関係を考慮したネットワークとして表現し、これを直接構文木の一つの分岐に割り当てることにより、次に形成される範疇の予測の度合や、新たに形成された構造から予測されていた範疇への到達可能性を活性化レベルにより示すことができる。一方、人間の言語運用における資源制約の顕著なものとして短期記憶の有限性に注目し、すべてのユニット間に弱い抑制性リンクを設け弱い相互抑制効果を導入することにより、これに相当するものを実現している。そして、この構文解析機を用いて、人間の構文認知過程に現れる興味深い現象である、ネストの深い埋込文や garden path sentence の認識困難、構造的な多義文の偏好現象などの解析を行った。

1. はじめに

人間の言語理解・生成モデルの研究の大きな柱の一つとして、人間の構文解析過程のモデル化に関する考察がいくつか行われてきた。人間が実時間で構文解析可能であるという知見の下、より高速な構文解析アルゴリズムを求めて研究が行われた。例えば、ATN, left-corner parser, chart parser, shift-reduce parser などが挙げられる¹⁾。

一方、人間の構文認知における特徴的な現象に注目した研究も行われている。これらの現象が人間の構文解析過程の効率の良さを解明する鍵となると考えられるからである。興味深い現象としては、例えば、ネストの深い埋込文や garden path sentence の認識困難、構造的な多義文の偏好現象などがある(各現象に関しては第4章を参照されたい)。このような現象を説明するために構文解析アルゴリズムに、現象に即したヒューリスティックな知識を埋め込むことが行われている。例えば、Marcus の wait-and-see parser や Shieber の shift-reduce parser など構文解析アルゴリズム自身に言及したモデルや、従来の構文解析アルゴリズムに、いくつかの“原則”を組み込むことにより実現するものもある²⁾。しかし、いずれの構文解析モデルにおいても、先に挙げた現象の一部分を説明で

きるにすぎない。

われわれは、これらの現象が人間の動的言語運用に纏わるものであり、文脈自由文法などにより生成される文をすべて扱うような静的逐次処理機構ではモデル化が難しいと考える。そこで、文法などで静的に記述しきれない、言語の動的運用面を解析するためにネットワークによる超並列計算モデルである Connectionist Model (以下、CM と略す)³⁾による構文解析モデルを構築した。本構文解析機では CM におけるユニットを文法範疇に割り当て、活性化レベルを句構造に対する注目度の高さと見なす。さらに、語が入力される度に句構造が動的に形成されていく過程を示すために動的結合機構を導入した。Waltz ら⁴⁾あるいは田村ら⁵⁾の自然言語理解モデルや Fan y の構文解析機⁶⁾など、従来の CM 構文解析機では静的に結合された構造間の活性化レベルの差のみに注目していたが、この機構により、動的に形成された構造の結合の強さを活性化レベルにより示すことができ、より詳細な議論ができる。また、句構造規則に相当する、活性化の順序関係を考慮したネットワークを、直接、構文木の一つの分岐に割り当てることにより、次に形成される範疇の予測の度合や、新たに形成された構造から予測された範疇への到達可能性を活性化レベルにより示すことができる。

2. Connectionist Model³⁾

CM は、互いに結合された多数の単純なプロセッサ(ユニットと呼ぶ、図では○で表す)が並列に計算を

† A Parsing Model Based on Connectionist Model by TATSUNORI MORI and HIROSHI NAKAGAWA (Department of Electrical and Computer Engineering, Faculty of Engineering, Yokohama National University).

†† 横浜国立大学工学部電子情報工学科

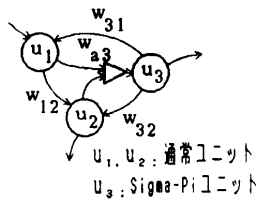


図 1 Connectionist Model
Fig. 1 Connectionist Model.

行う計算モデルである。各ユニットの内部状態は活性化レベルと呼ばれる値により表現される。ユニット間には活性化レベルのやり取りを行うための重み付きリンク（図では矢印で表す）が張られている。ユニット u_i の活性化レベルを $a_i(t)$ 、出力関数を f_i 、そして、 u_i から u_j へ向かうリンクの重みを w_{ij} とすると、ユニット u_i の次時刻の活性化レベル $a_i(t+1)$ は

$$a_i(t+1) = \sum_j w_{ji} \cdot f_j(a_j(t)). \quad (2.1)$$

で計算される。重み w_{ij} が正值のリンクを興奮性リンクと呼び、負値のリンクを抑制性リンクと呼ぶ。また、Sigma-Pi と呼ばれるユニットでは、入力として他の複数のユニットの出力の積和 $\sum w_{ji} \prod f_j(a_j)$ をとるので、ゲート素子のように使える。Sigma-Pi ユニットは他のユニットの活性化状態によりリンクの重みを変化させることができるユニットと考えられる（図では白抜ききの三角形により“積”を表す）。

CM は、神経回路網の研究の成果を認知過程の並列処理モデルとして応用したものであるが、われわれの立場としては、本稿で述べるメカニズムがそのまま人間の脳の中に“物理的に”存在するとは考えない。むしろ、CM を“注目点”の移り変わりを記述する枠組みを与えるものとして捉える。すなわち、活性化レベルは注目度を表し、ユニットのネットワークは、注目点の流れを方向付ける枠組みとして捉える。

3. Connectionist Model を用いた構文解析

われわれは、人間の構文解析過程に次の性質があるものと仮定している。

『語を読み込む度に、得られた情報をすべて使用し遅延させない。さらに、構成可能な構造は即座に構成してしまう。』

これは、人間の言語処理の効率の良さの理由として自然に導かれるものと考えている（文献 6）の議論が興味深い。この仮定より、構造的に曖昧な構文は並列に扱うことが導かれるが、これを支持する実験結果が

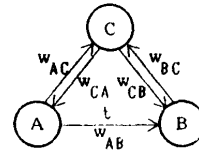


図 2 句構造ネットワーク
Fig. 2 A phrase structure network.

MacKay らにより報告されている⁴⁾。人間の言語処理において、曖昧な言語に対しては負荷が増大するというものであるが、これは、人間が曖昧な構文に対して複数の読み方を行っていることを示唆するものである。また、人間は通常、曖昧な文に対して同時には唯一の解釈を思い浮かべているが、複数の解釈間で行き来は容易である⁴⁾。以上より、曖昧な構文に対しては複数の可能性が互いに排他的となっていて、その内の一つだけが意識の上に乗るというモデルが考えられる。本論文で述べるモデルでは統語構造を表現するネットワークの活性化レベルに注目し、

『曖昧な構文に対しては、活性化の大きい構造が偏好される。』

と仮定する。これは、活性化レベルが注目度に相当することから自然に導かれると考える。

3.1 Connectionist Model 上での構文木の表現

CM において従来の“記号”に相当するものを表現する方法がいくつか提案されているが、ここでは局在表現 (local representation)³⁾を採用する。すなわち、CM のユニットに文法上の範疇を割り当て、そのユニット（範疇ユニット）の活性化が該当する範疇の形成に対応するとしよう。範疇ユニットによる物理的ネットワークが構文木を表現する。ネットワークにより直接、構文解析を行う場合、句構造規則に相当するものを表現しなければならない。例えば、句構造規則

$$\langle C \rangle \rightarrow \langle A \rangle, \langle B \rangle \quad (3.1)$$

は、

- (a) 範疇 A, B 間の順序関係
- (b) 範疇 A, B と範疇 C の依存関係

を記述しているが、これらは次のように表現する。(b)の“依存関係”については、興奮性リンクによりユニットの活性化の依存関係として表現できる。(a)の“順序関係”に関しては、活性化の順序関係を表現するリンクを新たに導入する。このリンクは、直後の範疇ユニットの活性化可能性に対してトリガをかける働きをすることから、トリガリンク (trigger link) と呼び、“ \rightarrow ”で表す。(3.1)式をトリガリンクを用いて表現すると図 2 のようになる。これを、句構造ネット

ワークと呼ぶ。トリガリンクはユニットAが“十分”活性化されていなければユニットBが活性化されていないことを保証する。

この図において重み $w_{CA}(w_{AC})$, $w_{CB}(w_{BC})$ は1より小さい正値で、それぞれ、範疇Cに対する範疇A、範疇Bの重要度を表している。また、重み w_{AB} も1より小さい正値で、範疇Aから範疇Bが予測される度合を示している。

トリガリンクは通常のユニットでは表現できないが、Sigma-Pi ユニットを用いれば実現できる。Sigma-Pi ユニットで構成したトリガリンクを用いた句構造ネットワークは、概略、図3のようになる。

3.2 構文木形成のための基本機構

3.1 節で構成した句構造ネットワークを構文木の一つの分枝として結合することにより構文木を成長させれば、構文解析を行うことができるわけだが、本節では、この時に必要となる基本メカニズムについて述べる。

3.2.1 句構造ネットワークのコピー

句構造ネットワークを構文木の構成要素として用いると、同じ構造の句構造ネットワークが複数必要になる。例えば、句構造規則が再帰的に定義されている場合などである。最も簡単な対応方法は、あらかじめ同じ句構造ネットワークを複数用意することである。しかし、この方法では句構造規則の学習過程をうまく論じることができないという問題が生じる。

学習については、本論文では詳細を議論しないが、興味深い問題の一つである。句構造規則の学習は、句構造ネットワーク内の範疇ユニット間の結合パターンの重みの学習に相当する。同一の句構造規則に対して複数の句構造ネットワークを用意すると、句構造ネットワークを利用することによる学習の効果が分散してしまい、すべての句構造ネットワークの学習が均一であることが期待できず、句構造規則に関する学習結果

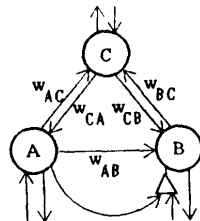


図3 Sigma-Pi ユニットによる句構造ネットワーク
Fig. 3 A phrase structure network based on Sigma-Pi units.

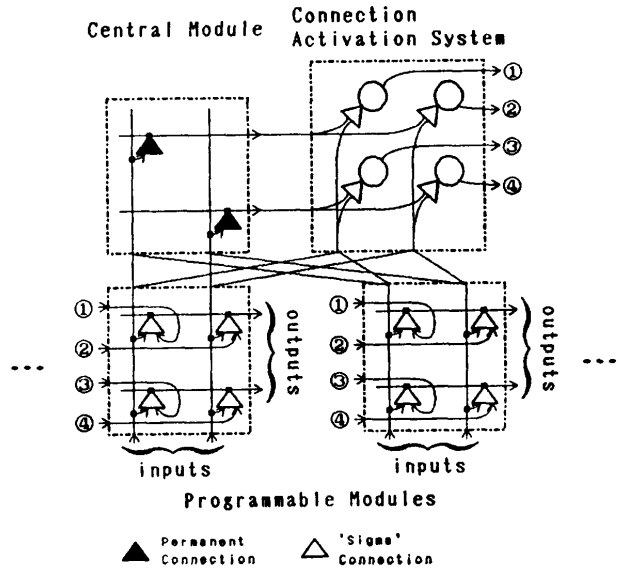


図4 簡単なCIDの例
Fig. 4 A simple example of the connection information distribution mechanism.

はどのように蓄積されていくかをうまく説明できない。

そこで一つの句構造規則に対して、学習を行う句構造ネットワーク(オリジナル)を一つだけ用意し、構文木を構成するときには、これをコピーして用いることを考える。具体的なコピー機構は **Connection Information Distribution mechanism (CID)**⁹⁾ を応用している。CIDは、学習により形成される恒常的な結合を持つネットワーク (CMod: Central Module) から、Central Module と同じ構造を持つ各結合が可変で制御可能な“白紙”のネットワーク (PM: Programmable Module) に結合の情報だけを転写する。PMはSigma-Pi ユニットの積結合を用いて構成されている。図4に簡単なCIDを示す。

この図において一つのPMに注目し、その下方左側の入力に活性化レベルが伝播してきた場合を考える。PMの入力①~④は、それぞれ、PM内の積結合の一方となっているので、ここへ伝播される活性化レベルがPM内の結合の重さと等しくなる。初めは、①~④へは入力を与えられないので、PM内の入出力間の結合の強さは0である。そのため、入力された活性化レベルは出力側に伝播されることなく、CModの下方左側入力および、Connection Activation System (CAS)の下方左側入力に伝えられる。CMod内の転写すべき恒常的な結合に従って、CMod下方左側への入力は右上の出力へのみ伝播される。CASでは、CMod

の出力とPMの出力を用い、結合情報の抽出が行われる。今の場合、CASの左上と下方左側の入力に活性化レベルが伝播してきたので、CAS内の左上のユニットが活性化され、①へ活性化レベルが伝播してくる。①へ伝播される活性化レベルはPM内の左上の結合の重みに等しいので、この結合が形成されたことになり、PMの下方左側の入力は右上の出力に伝播される。これは、PMの下方左側の入力に關与する結合をCModから転写してきたことに相当する。

3.2.2 コネクタ—動的結合機構

語が入力される度に構文木を動的に成長させるためには、動的にリンクを作り出す機構が必要である。CMには動的にリンクを生成する枠組みはないが、Sigma-Piユニットにより動的にリンクの重みを変化させることはできる。通常は重みが0であるリンクを、あるユニットの活性化状態により変化させて正值の重みとすれば興奮性リンクを生成したことになる。この機構を用いた動的結合機構をコネクタ(Connector)と呼ぼう。コネクタの基本動作は、ある活性化したユニットを他の活性化可能なユニットに結合することである。これを、スーパーバイザなしに行うために、結合要求用のコネクタと結合受理用のコネクタを用意する。それぞれの動作は、概略、次のとおりである。

(a) 結合要求型コネクタ

自分の所属するユニットが活性化されると、自分の受け持つすべての結合受理型コネクタに結合要求信号を送り、結合受理信号を待つ。なお、これらの信号はリンクを介した活性化伝播として表現する。結合受理信号が送られてきた場合には、それを送ったコネクタとのリンクを確立する。

(b) 結合受理型コネクタ

自分の受け持つ結合要求型コネクタからの結合要求信号を待つ。結合要求信号を受信したときに所属するユニットが活性化可能で、結合要求信号が“受理条件”を満たすならば、結合要求信号を送信した結合要求型コネクタに結合受理信号を送信する。

結合受理型コネクタに埋め込まれた“受理条件”が動的結合のアルゴリズムに相当する。

3.3 構文解析アルゴリズム

本構文解析機は

『句構造ネットワークに埋め込まれた句

構造規則を表す結合パターン』

『複数の句構造ネットワークを動的に結合するコネクタ』

によりアルゴリズムが記述されている。

ここでは句構造規則として

$$\langle C \rangle \rightarrow \langle A \rangle, \langle B \rangle \tag{3.2}$$

$$\langle D \rangle \rightarrow \text{Word} \tag{3.3}$$

の形式を持つ文脈自由文法(チョムスキー標準形)だけについて考える。〈A〉、〈B〉等は文法範疇、Wordは語彙中の一つの単語である。便宜上、(3.2)式において〈A〉の位置にある範疇を総称して範疇A、句構造ネットワーク上で〈A〉に相当するユニットを総称してユニットAと呼ぶ。〈B〉、〈C〉についても同様である。また、〈D〉は〈C〉と区別せずに扱う。

構文解析のためのネットワークは図5のように、オリジナル句構造ネットワーク、これをコピーして用いる“白紙”の句構造ネットワーク、および、辞書に相当するネットワーク群より構成される。

本構文解析機への語の入力は、これに対応する範疇ユニットへの外部入力として表現する。この外部入力は、対応する語が入力されたとする時点から与えられるが、常に一定の活性化レベルを保つのではなく、ある時定数で減衰することを仮定している。これは、われわれが文を読んだり、聞いたりする場合には、知覚したすべての語に相当する刺激(視覚、聴覚)が一度に与えられるのではなく、ある幅を単位として刺激が知覚され、新たな語(あるいはそれよりも小さい単位

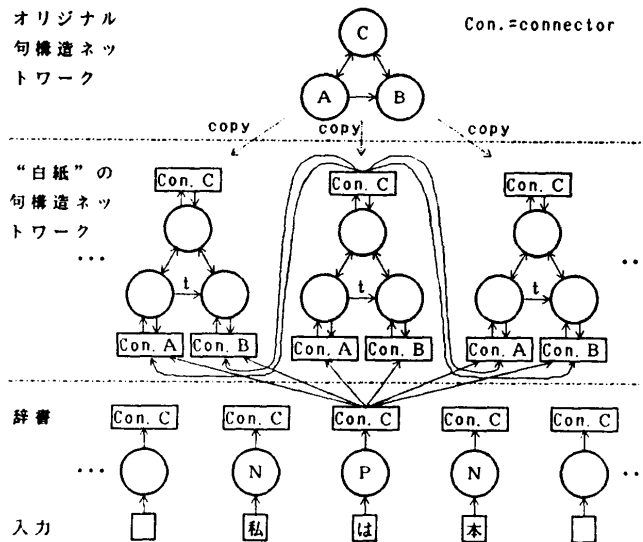


図5 構文解析ネットワークの概観
Fig. 5 The overview of the parsing network.

である文字・音素)が刺激として与えられる度に知覚の焦点が移行すると考えられるためである。

句構造ネットワークを動的に接続するために各ユニットに対応して三種類のコネクタ, A (結合受理型), B (結合受理型), C (結合要求型)を用意する(図5参照)。なお詳細は述べないが, 図5において便宜上 A, B, C など単一範疇ユニットにより表現しているものは, 実際には複数の範疇の可能性を網羅したネットワークである。また, いずれのコネクタも Sigma-Pi ユニットのネットワークとして実現される。

(1) コネクタ C (結合要求型) のアルゴリズム

コネクタ C が結合要求を送信するのは,

(a) 『単語が入力され, その範疇ユニットが活性化された場合』(辞書部)

(b) 『範疇ユニット A, B の活性化により範疇ユニット C が活性化された場合』(句構造ネットワーク部)

である。この結合要求は他の句構造ネットワークのコネクタ A およびコネクタ B に送信される。複数の結合が確立した場合には, それぞれの結合は互いに排他的に作用する。すなわち, 結合受理を返信したコネクタを持つ範疇ユニットの活性化レベルに応じて結合の強さが変化し, 結合の強いものが他の結合を妨げるように働く。これは本章冒頭で述べた曖昧な構文の取り扱いの方針に従うものである。(以降の図では破線矢印“→”で, この抑制性を表す。)

また, (b) についてはユニット C の出力関数の閾値の設定により方策が変わるが, 閾値が比較的低い値を持っていて, ユニット A が活性化されただけでコネクタ C の結合要求信号が送信されるとする。これは未完成の構造からも, 一歩先に, より上位の構造を構築してしまうという方策になるが, 予測された範疇への到達可能性のチェックを含んでいる点に注意されたい。すなわち, ある語が入力されると, これに対応する範疇から, この範疇を最左の葉に持つ上位構造が順次上方に向けて構成されるが, この時, 構成された上位構造のうち, 予測範疇としてトリガされている範疇と結合できるものは, 将来適切な構造を構成しうるものとして, 予測範疇からの活性化レベルのフィードバックを受けられ, その構造全体の活性化レベルが強化される。一方, 予測範疇に結合できない構造は, 予測範疇からの活性化レベルのフィードバックを受けられないままである。このように, ある語が入力されると, その語から予測された範疇に至る経路を含む部分的な木

構造が構成され, 活性化レベルの強化を受ける。なお, 次節(2)において構文解析例をもとに, 到達可能性チェックについて述べているので参照されたい。

(2) コネクタ A (結合受理型) のアルゴリズム

コネクタ A は他の句構造ネットワークのコネクタ C から結合要求信号を受信する。コネクタ A の受理条件は次を満たすことである。

『結合要求を発している句構造ネットワークの範疇と同じ範疇を範疇 A に持つオリジナル句構造ネットワークを, 結合要求を受け取った“白紙”の句構造ネットワークにコピーできた。』

(3) コネクタ B (結合受理型) のアルゴリズム

コネクタ B も, 他の句構造ネットワークのコネクタ C から結合要求信号を受信する。受理条件は次を満たすことである。

『結合要求を受けたコネクタ B を持つユニット B がトリガされていて, 活性化可能である。』

予測された範疇 B に対して複数の構造の可能性が存在する場合には, それらが互いに排他的となるように抑制性リンクにより互いを接続する。これも, 曖昧な構文の取り扱いに関するわれわれの方針に従うものである。(図では “[...]” によりユニット間の排他性を表す。)

3.4 構文解析例

本節では,

『私は本を読む。』

を例文にして, 構文解析の様子を追ってみよう。

構文解析の完了したネットワークを図6に示す。

(1) “私は” が入力された直後

ユニット PP₁ が活性化されるので, PP₁ のコネクタ C が結合要求を他ユニットのコネクタ A, B に送信する。現在, トリガされて活性化可能なユニット B はないが, 範疇 PP を左部分木に持つ句構造が存在するので, これを, “白紙”, つまり, 使用されていないプログラム可能な句構造ネットワークにコピーすることができる。コピーされた句構造ネットワークのコネクタ A は結合受理信号を先のコネクタ C に返信し, 結合が確立される。一方, ユニット PP₂ は十分に活性化されるので, 次範疇に相当するユニット VP₁ は活性化可能状態になる。また, 上位範疇に相当するユニット VP₂ も活性化され, そのコネクタ C は結合要求を送信するが, 現在活性化可能なユニット VP や, VP を左部分木に持つ句構造がないので, 結合は確立されない。

(2) “本を” が入力された直後

(1)と同じ過程で、 VP_3 を根とし VP_4 が活性化可能な状態にある構造が構成されるが、 VP_3 自身は未完成である。前節(1)で述べたように、このような未完成の構造からも結合要求を行うことができるが、(1)においてユニット VP_1 が活性化可能状態となっていたので、 VP_3 のコネクタCからの結合要求は VP_1 のコネクタBに受理され結合が確立される。ここで、もし、未完成の構造から結合要求ができないとすると、 VP_3 は、右部分木に相当する構造がすべて構成された後に、はじめて結合要求ができることになり、 VP_3 が予測された範疇と一致しない場合には、かなりの作業を無駄に行うことになる。

(3) “読む”が入力された直後

VP_3 のコネクタCからの結合要求が VP_4 のコネクタBに受理され、結合が確立し、構文解析木に相当するネットワークが得られる。

3.5 場の抑制による短期記憶モデル

ネストの深い埋込文や garedn path sentence の認識の困難さなどには、人間の言語運用における何らかの資源制約が大きな役割を果たしていると考えられる。もし言語認識過程ですべての可能性を保持できるならば、これらは正しく認識できるからである。ここでは、言語運用に対して最も影響を与えると思われる短期記憶の有限性について考える。

われわれ人間の短期記憶の容量は、通常、 7 ± 2 チャンクであることが知られている。チャンクとは記憶の単位となりうる、記憶の“塊”である。短期記憶の有限性は、記憶単位間に相互抑制作用が存在するため

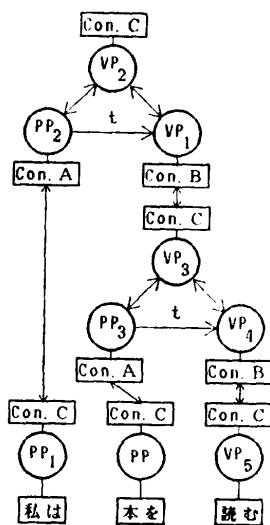


図6 構文解析例

Fig. 6 An example of the network parsing tree.

と考えられる。

CMにおいて、このような短期記憶は次のようにモデル化できよう。

(a) 通常用いている結合リンクは、記憶において相互作用の強い部分を明示的に示したものであり、これが短期記憶における一つのチャンクを構成する。

(b) 各チャンク間の相互抑制効果を表現するために、全ユニット間に負定数の重みを持つリンクが暗に存在すると仮定する。

(b)のリンクを隠れリンク (hidden link) と呼び、その効果を『場の抑制』と呼ぼう。各ユニットの出力が活性化レベルの線形閾値関数の場合は、場の抑制効果が出力関数の閾値の増加に相当する。活性化したユニットが増加すると、場の抑制効果により各ユニットの閾値が上昇するので活性化が抑制される傾向が強くなる。たとえば、構文解析過程で並列に存在する複数の構造は、活性化レベルの低い構造から淘汰されていく。

4. 種々の言語現象に対する本モデルでの解析

本章では、第3章で構成した構文解析モデルを用いて、いくつかの興味深い言語現象の解析を行う。

4.1 ネストの深い埋め込み文に関する解析

本モデルでは短期記憶の有限性に原因を求め、認識の困難さをモデル化できるが、次の文で示すように、短期記憶の有限性だけでは説明しきれない場合についてもモデル化ができる。

“The man who the girl who the dog chased liked laughed.”

この文³⁾は、さほどネストが深くないにもかかわらず、後半部まで読んだところで困難を感じる (liked や laughed の主語が、すぐにはわからないであろう)。主語となるべき名詞はすべて、記憶に残っているにもかかわらず、動詞との対応がはっきりしなくなる。解析のために“liked”が入力される直前のネットワーク表現の構文木を描いてみると図7となる。なお、コネクタおよび説明に不要なユニットは省略する。“chaded”が入力された直後では、 VP_2 と VP_1 が予測範疇として未完成のままとなっている。次に入力される動詞“like”の範疇ユニットはコネクタによって、このうちのいずれかと接続されるのだが、これは、 VP_2 および VP_1 に対する予測の度合、すなわち、

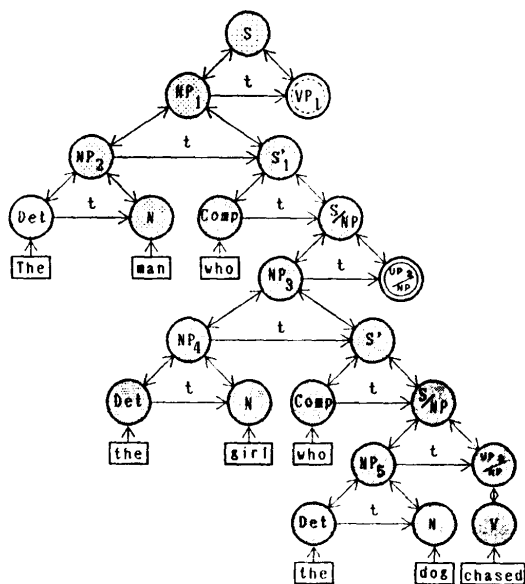


図 7 “liked” が入力される直前のネットワーク
Fig. 7 The network parse tree immediately before inputting “liked.”

直前の範疇ユニット NP_3 , NP_1 の活性化レベルによるトリガの割合によって決定される。ここで、 NP_3 と NP_1 の活性化レベルを比較してみる。 NP_3 は “chased” が入力されたことにより構造としては完成したが、 NP_3 の主要構成素である NP_4 が完成してから埋め込み文一つ分の作業を行っているため、入力から伝播される活性化レベルが時間経過により減衰していること、および、場の抑制効果により、 NP_3 の活性化レベルは低くなりつつある。ここで、句構造ネットワークにおける主要構成素からの活性化伝播の寄与分、すなわち、主要構成素と上位範疇の間の結合の方が、その他の構成素のものよりも大きいことを仮定している。概略、 NP_3 の活性化レベルは、 NP_4 からの活性化伝播によって支えられているのだが、 NP_4 を根とする句構造ネットワークでは、これを構成する語 “the” “girl” に対応する入力活性化レベルが減衰してしまっており、このネットワーク内でのフィードバックによってのみ活性化レベルを維持している。さらに、この句構造ネットワークは場の抑制効果を受けている。

一方、 NP_1 を見ると、その主要構成素である NP_2 は既に活性化されており、また、 S'_1 もほとんど完全に近いが、 NP_3 の場合と同じ理由で活性化レベルが低くなりつつある。ここで、 NP_1 を支える主要構成素 NP_2 に注目すると、 NP_3 の主要構成素であった NP_4 とネットワークとしては同じ構造を持ち、しかも、これを

構成する語に対応する入力活性化レベルも同じように減衰しているので、 NP_4 とほぼ同じ状態にあることがわかる。

以上より、 NP_3 と NP_1 はほぼ同様の活性化レベルにあり、明らかな差はなくなっている。このため、“liked” が入力されたときに、未完成の予測範疇のうち、正しい構造である VP_2 が、すぐさま選択されるとはいえなくなってしまう。つまり、『主語と動詞の対応が容易にとれない』状態を示している。これは、“chased” が入力される時に、 VP_3 の直前の範疇が完成されたばかりで、しかも、入力の活性化レベルがほとんど減衰していないので、活性化レベルが高く VP_3 と VP_2 の予測の割合に大きな差があり VP_3 を容易に選択できたことと対照的である。

4.2 多義文の偏好現象に関する解析

従来の記号処理パラダイムにおける解析では、多義文の偏好現象を、右連合（遅延閉包、低位付加）原理、最小（発動的、高位）付加原理、語彙的偏好原理など、“原理” をアドホックに導入することにより説明することが多い^{2),7)}。しかし、このような“原理” を個別に陽に構文解析機に組み込むのは、原理間の整合性に関する問題を解決しない限り難しい。本項では、本構文解析機を用いて多義文の偏好現象を活性化レベルという一つの尺度だけを用いて論じる。

次の文を考えてみよう。

(1) 『花子は急いで私が選んだ本を買った。』

この文⁷⁾は、副詞 (Adv) “急いで” の修飾の仕方に関して、構造的に曖昧な文で、

(a) 『急いで～選んだ…』

(b) 『急いで～買った。』

の2通りに解釈可能である。このように構造的に曖昧な文に対し人間は一方の解釈を優先することが知られている。実際にアンケート調査（紙面に読点なしの文を記したものを黙読）を行ってみると (b) の読みを圧倒的に支持している（ただし、文献 7) は逆の報告をしているので疑問は残る）。解析のためにネットワークを描いてみると図 8 となる。“選んだ” が入力された直後では、 VP_{1b} および VP_{2a} が選択され、『花子は急いで私が選んだ。』という文が認識される。次に“本” が入力されると、その範疇 NP が活性化される。この時点では、 NP_a および NP_b がトリガされていて活性化可能であるから、これら二つの範疇と今活性化された NP “本” との結合が確立される。3.3 節に述べたように、動的結合の強さは結合先ユニットの活性

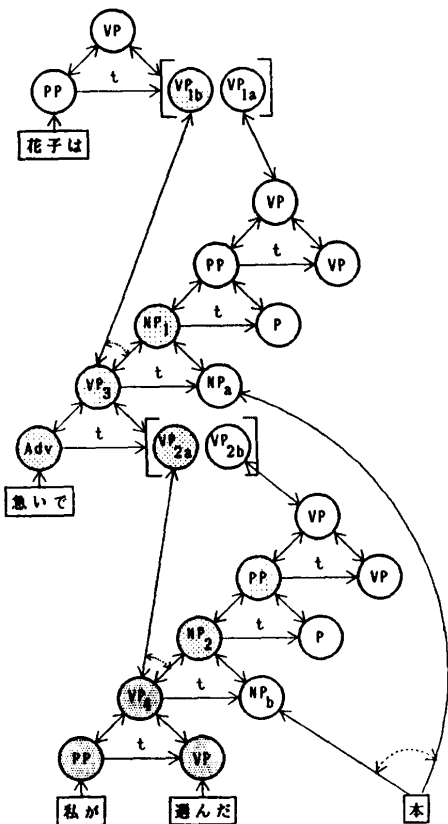


図8 “本”が入力された直後のネットワーク
Fig. 8 The network parse tree immediately after inputting “HON.”

化レベルに応じて決まり、候補が複数存在する場合には互いに結合の重みを減らすように抑制しあうが、NP “本” からの活性化伝播は同じなので直前範疇からの活性化伝播の差を調べよう。VP₄ は “本” を入力する直前に完成したばかりであるから活性化レベルは十分高い。一方、VP₃ は VP₄ からの活性化伝播により活性化されており、左部分木である Adv は時間経過および場の抑制効果により、VP₄ に比べると活性化レベルが低い。よって、NP_b との結合が選択され、(b) が選択される。本構文解析モデルでは、新しく入力された範疇は直前に完成した構造のうち活性化レベルの高い構造と結合しやすいという “引き込み現象” が見られる。

この他、

- (2) 『花子は太郎と次郎を殴った。』
 (“花子は (太郎+次郎) を殴った” を偏好)
- (3) 『美しい彼女の歌声』
 (“美しい…歌声” を偏好)

など”も “引き込み現象” によりモデル化できる。

従来原理として用いられてきた低位付加原理や高位付加原理などは、修飾語に注目し、それがどの動詞句あるいは名詞句を修飾するかを表したものである。例えば、例文(2)では後置詞句 “太郎と” の付名位置が低位の構造が偏好され、例文(3)では形容詞 “美しい” の付加位置が高位の構造が偏好される。原理として高位付加と低位付加というように相反するものが現れ、それらを使い分けなければならなくなるのは、過去に現れた修飾語がどこに付加するかという観点から眺めているためである。

本構文解析機では、いずれも新たに接続される構成素に対する “引き込み現象” および “高活性化レベル構造の選択” (図中 [] 内の範疇の選択) による効果として、活性化レベルという統一した尺度で説明可能である。しかし、これらの効果が相反する場合があり、この時の活性化レベルの移り変わりを統語情報だけで議論するのは難しいかもしれない。例えば、

- (a) John carried the groceries for Mary.
(b) She wanted the dress on that rack.

は同じ構文であるが、(a)では PP が VP を修飾する読み (高位付加) が偏好され、(b)では NP を修飾する読み (低位付加) が偏好される。高活性化レベル候補の選択が優勢な場合が(a)で、 “引き込み現象” が優勢な場合が(b)に相当する。このような偏好現象を解析するには、格情報における整合性や意味的共起性などによる活性化レベルの移り変わりを考慮する必要があると思われる。特に格情報については、心的文法の実在性の問題と密接な関係があると考えられる。例えば、(a)では、VP→VP, PP, VP→V, NP、に対応するネットワーク間の結合が格構造からの期待として学習により強化され、VP→V, NP, PP、という文法に近いものを利用していると考えられる。

4.3 Garden Path Sentence に関する解析

次の文を考えてみよう。

“The cotton clothing is made of grows in Mississippi.”

この文は、 “of” までは順調に読み進めることができるが “grows” を聞いたとたんに理解困難に陥る (正しくは、 “of” までが主語で “clothing…of” は関係節)。このように、文の途中まで、ある解釈で容易に解析可能であるが、その解釈が正しくないと認識された時点で、別の正しい解釈に移行することが困難であるような文は garden path sentence と呼ばれる。Marcus

は、このような garden path sentence が理解困難に陥る原因として、『人間が文を読み進む過程では解釈を一意に決定している。』とし、途中までに決定した仮説的な構造を捨てなければならない garden path sentence に対して人間は強い認知的困難を感じるとしている⁸⁾。

しかし、この文は常に袋小路に入ってしまうわけではない。例えば次のような場合である⁴⁾。

“Cotton is grown in several of the Southern States.

The cotton clothing is made of grows in Mississippi.” (4.1)

つまり、文脈の効果の入る余地を残しつつ、単文の場合の認知困難について論じる必要がある。“grows”の入力される直前のネットワークを図示したものが図9である。ただし、トリガリンクを省略している。

図をみるとわかるように、“cotton”が入力されると、その範疇の多義性に応じて範疇 Mod (修飾語) と範疇 NP が活性化される。そして、次範疇として NP と S' (補文) を予測する。“clothing”を入力すると、“clothing”から、これら予測された範疇への到達可能性が調べられる。ここで、両者の到達可能性の高さを比較すると圧倒的に NP への到達可能性が高い。これは、

“clothing”自身が NP であることと、関係節において S' の主語が欠損する場合に関係代名詞を明示しない場面が少ないことから、S' への到達可能性の根拠となる経路 [NP→S/NP→S'] のリンクはあまり学習されず、重みが小さいことによる。このために、NP_a が選択され、補文構造はほとんど構成されない。“grows”を読むと、NP_a の構造が却下されるが、もう一つの候補である補文構造はほとんど構成されていなかったの、選択し直すことはできず、理解困難に陥る。

次に、本構文解析モデルに文献 4) などに述べられているような簡単な文脈処理モデルを付け加えた場合を想定してみる。文献 4) には、文脈処理の簡単なモデルとして、構文木ネットワークからその中で使われた語の多義性など語彙レベルの情報を活性化レベルとして抽出し、また、大まかな文脈を捉えるために一つの文脈設定の一つのユニットの活性化に対応させる方法が述べられているのだが、これと同様の機構を仮定して、(4.1)式のように文脈が付加された場合について考えてみる。(4.1)式の一つめの文は、名詞句としての“cotton”を活性化する働きがあるので、この文が garden path sentence の前に置かれている時には、図9の構文解析過程において“cotton”の範疇の多義性に関して、修飾語 Mod よりも名詞句 N をより強く活性化する。このバイアス効果は S' の予測を促進し、補文構造が構成されることを促す。このように文脈により補文構造の選択を促すことができる。

5. おわりに

本稿では CM による構文解析モデルを提案し、種々の言語現象を活性化レベルの流れにより統一的にモデル化できることを示した。すなわち、CM のネットワークにより句構造規則に相当するものを表現し、動的結合機構により、これらを結合するというメカニズムにより、構文解析の動的過程を活性化レベルの流れとして統一的に表せることを示した。さらに、本構文解析モデルを用いると、ネストの深い埋め込み文や garden path sentence の認識困難、構造的な多義文の偏好現象など、人間の構文解析における特徴的な現象をいくつか解析できることを示した。

しかし、文脈自由言語の枠外にある自然言語現象については、本モデルの枠組みでは不十分

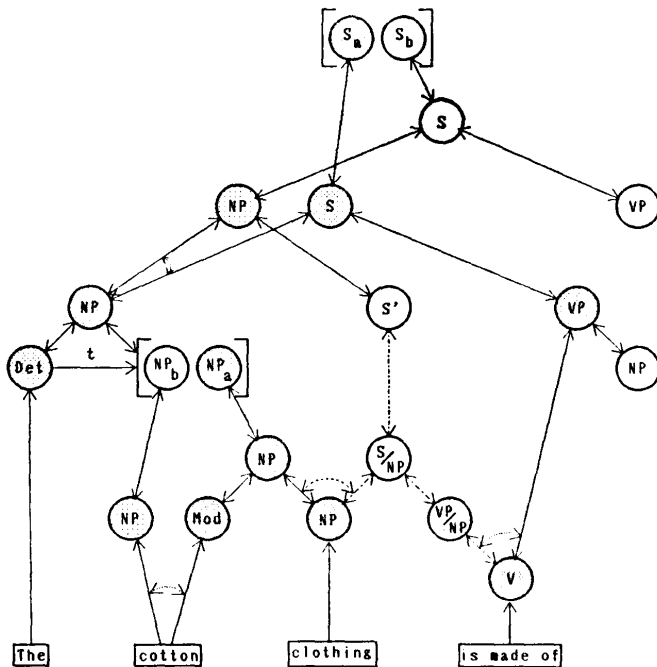


図9 “grows”が入力される直前のネットワーク
Fig. 9 The network parse tree immediately before inputting “grows.”

である。このような現象の一つとして非有界依存現象（長距離依存現象）がある。非有界依存現象は、統語構造の二つの位置の間に関係が存在し、しかも、途中でどんな大きい部分木が介在しても良いという現象である。例えば、関係節などに見られる空範疇と関係代名詞の間に関係などは、この現象の一つである。このような現象をモデル化するには、本モデルを拡張しなければならないであろう。また、メカニズムについては、動的結合機構の構成法に問題が残る。本論文で提案した動的結合機構では、動的に結合を生成するのではなく、動的に結合の強さを変化させるものなので、あらかじめ、すべての結合の可能性を網羅しておく必要がある。この方式を採らず、結合を動的に生成する仕組みを構成するためには、動的結合を制御するスーパーバイザを設定せざるをえないであろう。従来の記号処理パラダイムと CM によるアプローチを融合する節点として、動的結合の問題が存在するのかもしれない。

謝辞 日頃、有益な議論を頂く人工知能勉強会 AIUEO の諸氏に感謝いたします。なお、本研究は文部省科学研究費「特定研究 言語情報処理の高度化(課題番号 62210004)」ならびに稲盛財団研究助成金より予算の援助を得ている。

参 考 文 献

- 1) 野村浩郷：自然言語処理の基礎技術，電子情報通信学会 (1988).
- 2) 辻井潤一：文解析方式，情報処理，Vol. 27, No. 8, pp. 924-932 (1986).
- 3) Rumelhart, D. E. et al.: *Parallel Distributed Processing*, The MIT Press, Cambridge(1986).
- 4) Waltz, D. L. and Pollack, J. B.: *Massively Parallel Parsing: A Strongly Interactive Model of Natural Language Interpretation*, *Cognitive Science*, Vol. 9, pp. 51-74 (1985).
- 5) 田村 淳, 安西 祐一郎: Connectionist Model を用いた自然言語処理システム, 情報処理学会論文誌, Vol. 28, No. 2, pp. 202-210 (1987).
- 6) Hasida, K.: *Bounded Parallelism*, Doctoral Dissertation, Univ. of Tokyo (1985).
- 7) 往住彰文: 日本語の構造的多義性を決定論的に解決するための一般的規則について, 日本認知科学会第2回大会論文集 (1985).
- 8) Marcus, M.: *A Theory of Syntactic Recognition for Natural Language*, MIT Press, Cambridge (1980).

(昭和 63 年 7 月 14 日受付)
(平成 元年 1 月 17 日採録)



森 辰則 (正会員)

昭和 39 年生。昭和 61 年横浜国立大学工学部情報工学科卒業。昭和 63 年同大学院工学研究科電子情報工学専攻博士課程前期修了。現在、同博士課程後期在学中。人工知能、自然言語理解などに関する研究に従事。人工知能学会、日本認知科学会各会員。



中川 裕志 (正会員)

昭和 28 年生。昭和 50 年、東京大学工学部電気工学科卒業。昭和 55 年、東京大学大学院博士課程修了。工学博士。昭和 55 年 4 月より横浜国立大学工学部勤務。現在、同工学部電子情報工学科助教授。人工知能の研究に従事。AIUEO, 電子情報通信学会, 日本認知科学会, 日本ソフトウェア科学会各会員。