

統合パーサによる統合的自然言語解析†

劉 学 敏^{††} 西 田 豊 明^{††} 堂 下 修 司^{††}

本論文では、統合パーサと呼ぶ統合的自然言語処理を実現するための新しいメカニズムを提案する。統合パーサは、整合性維持エンジン CME と確からしさ維持エンジン PME の二つのサブエンジンからなっている。CME は処理の過程で作られた言明または仮定の間の依存性を保持し、これに基づいて、言明が導かれた根拠の提示、仮定の変更に応じる言明の集合の変更、矛盾が検出されたときの仮定の切り換えなどを行う。一方、PME は仮定に関する確からしさ情報を信念ネットワークの形で動的に維持し、推論過程を最も確からしい方向に誘導する。我々はこの統合パーサを利用して、統合的自然言語解析システムを試作した。このシステムは、処理の過程で語彙解析、構文解析、意味解析などの異なるレベルの解析を並行して進め、各解析レベルの部分結果の統合によって、曖昧性を早く解消することができ、また、本質的に解消できないような曖昧性を含んだ文に対して、このシステムは文脈情報を利用して、最も確からしい解釈を見つけることができる。

1. はじめに

自然言語の曖昧性はそれが発生した解析レベルの情報だけでは解決できないことが多い。例えば、統語情報だけをを用いて文の統語構造を一意に決定できない。また、意味構造も統語や文脈に関する情報なしには決定できない。このため、従来の単語解析、構文解析、意味解析を逐次的に進めるといような処理方法では、解析の途中段階で爆発的に多くの曖昧性が出現する。

このような問題を解決するには、統合的な解析、すなわち、単語解析、構文解析、意味解析の三つのレベルの解析を並行して処理を行うということによって、曖昧性の早期解消、無駄な処理の回避、解析の最も確からしい方向への誘導などの処理が必要であると考えられる。しかし、統合的な処理をどのように効率的に実現するかが課題であった。

本論文では、統合的自然言語処理を実現するために、統合パーサと呼ぶ新しいメカニズムを提案する。統合パーサは、次のような二つの機能を融合して構成される推論管理機構である。

(1) 論理的な整合性の維持

自然言語の曖昧性を解消するためには、何らかの仮定を作らなければ処理が進められないことが多い。このため、自然言語処理システムを構築する際に、仮定に基づく推論、矛盾が起きたときの仮定の切り換え、異なった仮定に依存する推論結果の取出しなどを考え

なければならない。本研究では、これらの操作をひとまとめにし、整合性維持エンジン CME (Consistency Maintenance Engine) を構成した。CME によって、処理の過程で作られた信念の整合性を維持する。

(2) 処理の確からしい方向への誘導

統合的な処理を効率に行うためには、処理を確からしい方向に進めることが望ましい。このためには、仮定の確からしさに関する情報を利用したり、新しい推論結果を得たとき、以前に設定した仮定の確からしさを修正したりする機構が必要である。そこで、我々は信念ネットワーク²⁾の手法を導入して、確からしさ維持エンジン PME (Plausibility Maintenance Engine) を構成した。これを利用して、処理を確からしい方向に誘導する。

我々は上に述べた統合パーサを利用して、図 1 に示すような自然言語解析システムを構築した。このシステムでは、統語解析、構文解析、意味解析などの処理を並行して行い、各解析レベルの処理結果の統合によって、曖昧性の早期解決を行う。また、解釈が唯一に定まらない曖昧な文に対しては、システムは知識や文脈情報などによって、最適な解釈を導出することができる。本論文では、第 2 章で統合パーサの構成について述べ、第 3 章で統合パーサに基づく自然言語解析システムの実現について述べ、第 4 章で実験と評価について報告する。また、第 5 章では、本研究で提案する方法を他の関連研究と比較して検討する。

2. 統合パーサの構成

統合パーサは二つのサブエンジン CME と PME を融合して構成されている。

† Integrated Natural Language Analysis with an Integrated Parsing Engine IPE by XUEMIN LIU, TOYOAKI NISHIDA and SHUJI DOSHITA (Department of Information Science, Faculty of Engineering, Kyoto University).

†† 京都大学工学部情報工学教室

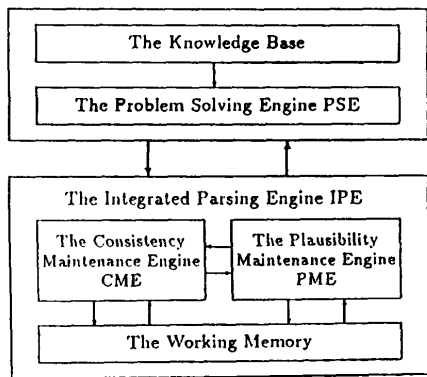


図1 統合パーサを用いる自然言語解析システムの構成
Fig. 1 The structure of the natural language analyzing system based on the Integrated Parsing Engine IPE.

2.1 整合性維持エンジン CME

整合性維持エンジン CME (Consistency Maintenance Engine) は、自然言語処理の過程で作られる言明の集合の論理的な整合性を維持し、処理の過程が矛盾に陥らないことを保証する。

2.1.1 CME と ATMS

CME は、de Kleer の ATMS (Assumption-based Truth Maintenance System)¹⁾ の技法をベースにして構成されている。ATMS と同じように、CME では、言明の内容にはかかわりなく、言明間の依存関係に注目した信念の管理を行う。言明は節点と呼ぶデータ構造を用いて管理される。節点には、仮定節点と非仮定節点の2種類がある。仮定節点は仮定を表す。非仮定節点は、仮定の組合せのもとで成立する言明を表す。仮定の組合せは環境と呼ばれる。各節点には、この節点が成り立つすべての環境を LABEL というデータ構造で記録する。節点間の依存関係は理由付け (justification) と呼び、リストとして記録する。CME は与えられた理由付けの集合を用いて、各節点の LABEL を推論の進行とともに動的に更新する。また、環境が与えられると、CME は節点の LABEL を用いて、その環境で成り立つすべての節点を求めることができる。

本来 ATMS は横型探索のための推論管理システムである。これをそのまま自然言語処理に適用すると、推論環境の爆発という問題が出てくる^{4),5)}。一般に、仮定の数を n とすると、可能な推論環境の数は 2^n となる。仮定が増えると、実質的にすべての環境を考慮するのは不可能である。

この問題を解決するために、我々の CME は節点の

集合を管理するだけでなく、仮定の組合せからなる環境を管理することも考える。これについて次に述べる。

2.1.2 推論環境の管理

自然言語処理の過程において生じる環境を分析してみると、設定される仮定が排他的であることが多いので、矛盾を含んだ仮定の組合せははじめから考慮に入れる必要はない。そこで、CME では、仮定の排他性を利用して、環境の木という技法を用いて、無意味な環境をはじめから作り出さず、矛盾に陥ることが判明していない環境 (これを推論可能な環境と呼ぶ) だけを取り出すようにした。

まず、特殊な節点として、仮定の組合せを表す節点 (これを環境節点と呼ぶ) を導入する。CME は環境節点を利用して、次のように環境の木を構成する。

仮定を全く含まない環境 $\{$ を環境の木のルート節点とし、この環境から処理を始める。処理の過程において、ある環境 E から曖昧性が生じ、排他的な k 個の仮定 A_1, \dots, A_k が作られると、CME は新たに作られた仮定の各々をもとの環境 E に加えて、新しい推論環境 $E_1, \dots, E_k (E_i = E \cup \{A_i\}, i=1, \dots, k)$ を生成する。これらの新しい環境をもとの環境の子孫環境と呼ぶ。その後の処理は子孫環境のどれかにおいて続ける。このように、CME はすべての推論可能な環境を一つの木構造の形に整理できる。また、環境の間の依存関係を記録するために、次のような理由付けを用いる。

$$E, A_i \rightarrow E_i \quad (i=1, \dots, k)$$

こうすると、環境 E あるいは仮定 A_i のどちらかが矛盾すると、それらに依存する環境 E_i も矛盾する。

環境の木の例を図2に示す。このように構成された環境の木は次の利点がある。

- (1) 排他的な仮定を含む環境が存在しないので、無駄な推論が回避できる。
- (2) 親環境は子孫環境の部分環境であるから、パーシングでは環境の木の葉節点だけを考慮すればよ

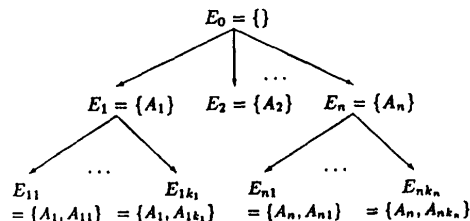


図2 環境の木

Fig. 2 A sample Environment Tree.

い。また、ある環境が矛盾することが分かれば、そのすべての子孫環境も矛盾に陥るので、探索範囲から除去すればよい。

(3) 推論は環境の木のすべての葉節点においても可能であるから、処理システムは横型探索と縦型探索の中間の任意の探索戦略をとることができる。

環境の木の各葉節点は可能な推論方向を指示している。CME は環境の木を利用して、推論システムの信念集合と処理方向を統合して管理し、処理過程の整合性を維持する。

2.1.3 CME による整合性の維持

CME は自然言語処理システムに対して次の機能を提供する。

- (1) 曖昧性に対応して仮定を作る。
- (2) 作られた仮定を基に、新しい環境を生成する。
- (3) 現在の環境で成り立つ信念の集合を決定する。
- (4) 矛盾が起きたとき、環境の木からすべての矛盾した環境節点を削除する。

処理システムはこれらの機能を利用して、CME に仮定を作らせたり、新しい推論環境を生成させたりする。例えば、処理の過程で、ある環境 E において、既存の言明 X_1, \dots, X_k から排他的に新しい結果 Y_1, \dots, Y_m が導出されると、システムはまず CME によって、仮定 A_1, \dots, A_m を作る。ここで、仮定 A_j は「節点 X_1, \dots, X_k から導出可能な結論は Y_j である」ことを意味する。次に、作られた仮定を基に、CME に新しい環境 E_1, E_2, \dots, E_m を生成させ、環境の木を拡張させる。そして、各推論結果 Y_j に対して、次のような理由付け

$$X_1, \dots, X_k, A_j \rightarrow Y_j \quad (j=1, 2, \dots, m)$$

を付加する。この結果、節点 Y_j が仮定 A_j を含む環境 E_j だけで成立するようになる。

解析の過程で矛盾が検出されると、処理システムは現在の推論環境が矛盾したことを CME に通知する。この場合、CME は節点の間の依存関係を利用して、矛盾した仮定の組合せを探し、環境の木からすべての矛盾した環境を削除する。

2.2 確からしさ維持エンジン PME

確からしさ維持エンジン PME (Plausibility Maintenance Engine) は仮定に関する確からしさ情報を信念ネットワーク (Belief Network)²⁹ の形で動的に管理し、推論過程を最も確からしい方向に誘導する。

2.2.1 信念ネットワークの構成

各仮定節点には、その仮定の確からしさを表す情報が付けられる。仮定の確からしさの設定はシステムの問題解決エンジン PSE (第3章参照) によって行われる。PME は仮定の確からしさと環境の木を利用して、次のように信念ネットワークを構成する。

まず、環境 E に対して、確信度 $P(E)$ という指標を導入する。 $P(E)$ は環境 E が矛盾しないことの事前確率を表す。(これに対応して、 $P(\neg E)$ で環境 E が矛盾することの事前確率を表す。ここで、 $P(\neg E) = 1 - P(E)$.)

環境の木のルート節点には何の仮定も含まれていないので、その確信度を1にする。ルート節点以外の節点 E_i の確信度はその節点の親節点 E の確信度 $P(E)$ と、 E と E_i の間の条件付き確率 $P(E_i|E)$ によって決められる。

$$P(E_i) = P(E) \cdot P(E_i|E)$$

ここで、 $P(E_i|E)$ は環境 E が成立するとき環境 E_i が成立することの確率である。これは仮定に与えられた確からしさによって決定される。例えば、ある環境 E で新しい仮定 A_1, \dots, A_n によって子孫環境 $E_1, E_2, \dots, E_n (E_i = E \cup \{A_i\}, i=1, \dots, n)$ を生成した場合、仮定 A_i の確からしさを p_i とすると、各子孫環境 E_i の親環境 E に関する条件付き確率を次のように定義する。

$$P(\neg E_i | \neg E) = 1, \tag{1}$$

$$P(E_i | \neg E) = 0, \tag{2}$$

$$P(\neg E_i | E) = 1 - p_i, \tag{3}$$

$$P(E_i | E) = p_i. \tag{4}$$

PME は上の p_i を環境 E_i に与えて、これを用いて、上の四つの条件付き確率に計算する。このように、PME は環境の木から一つの信念ネットワーク (実際は木構造) を構成することができる。例えば、図2で示した環境の木に対応する信念のネットワークは図3のようになる。ここで、 P_i, P_{ij} は仮定 A_i, A_{ij} の確

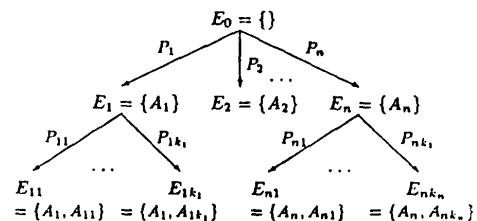


図3 信念ネットワーク
Fig. 3 A sample Belief Network (an Environment Tree with conditional probabilities).

からしさである。

PME はこのような信念ネットワークを利用して、環境の木から最も確からしい環境をシステムの現在の推論環境として選出する。これによって、推論過程を誘導する。具体的には、最も確からしい環境 E は次のように定義される。

- (1) E は環境の木の葉節点である。
- (2) 環境の木のルート節点から E に至る経路上のすべての節点はその兄弟節点より高い確信度を有する。
- (3) 上のような節点が数個あれば、その環境に含まれる仮定の数の多い方をより確からしいものにする。

2.2.2 確からしさの修正

仮定の確からしさはその仮定が生成された時点の処理結果に基づいて決定されたものである。処理の進行に伴って、新しい事象が生じると、確からしさを修正しなければならない。本研究では、現在の推論環境が矛盾したとき、確からしさの修正プロセスを起動する。

確からしさの修正は、ベイズの定理に従って行う。ある環境 E が矛盾したとき、 E 以外の環境 E_i に対して、その事後確率は次の式(a)によって計算できる。

$$P(E_i | \neg E) = \frac{P(\neg E | E_i) \cdot P(E_i)}{P(\neg E)} \\ = \frac{(1 - P(E | E_i)) \cdot P(E_i)}{(1 - P(E))} \quad (a)$$

PME はこの事後確率を利用して、環境の条件付き確率を修正する。例えば、環境 E_c の親環境が E_p である場合、環境 E が矛盾すると、PME は E_c の E_p に関する条件付き確率 $P(E_c | E_p)$ を次のように修正する。

$$P'(E_c | E_p) = \frac{P(E_c | \neg E)}{P(E_p | \neg E)} \\ = \frac{(1 - P(E | E_c)) \cdot P(E_c)}{(1 - P(E | E_p)) \cdot P(E_p)} \quad (b)$$

また、環境の確信度の定義によって、

$$P(E_c) = P(E_c | E_p) \cdot P(E_p)$$

であるから、式(b)は次のように変形できる。

$$P'(E_c | E_p) = \frac{(1 - P(E | E_c)) \cdot P(E_c | E_p)}{(1 - P(E | E_p))} \quad (c)$$

ここで、もし E_c が E の祖先節点であれば、 $P(E | E_p)$ と $P(E | E_c)$ は節点 E_p と E_c から節点 E までの経路上の条件付き確率から計算できる。一方、 E_c が E の

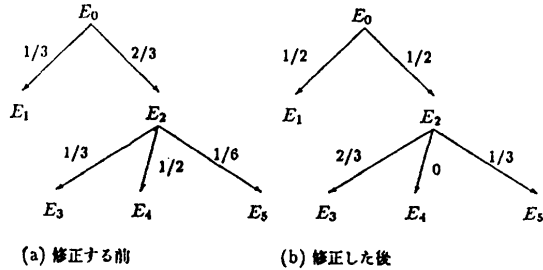


図 4 信念ネットワークの修正
Fig. 4 Updating the Belief Networks.

祖先節点ではないならば、仮定の排他性のため、 E_c と E が同時に成立することはない。すなわち、

$$P(E | E_c) = P(E | E_p) = 0.$$

したがって、この場合、 $P(E_c | E_p)$ を修正する必要はない。

例えば、図 4 (a) に示す環境の木において、もし環境 E_{11} が矛盾すれば、確からしさを修正した後の環境の木は図 4 (b) のようになる。

3. 統合パーサによる自然言語解析

統合パーサを用いる自然言語解析システムにおいて、その解析処理は問題解決エンジン PSE (Problem Solving Engine) の制御下で動作する。

PSE は自然言語解析のための各解析レベルで必要な解析操作(単語解析操作、構文解析操作、意味解析操作)とこれらの操作を制御するための制御部、そして、仮定の確からしさの設定機構からなっている。入力文を与えると、PSE は自然言語処理の各レベルの解析を並行して行う。処理の過程で作られた言明や仮定はすべて CME によって管理される。また、仮定に関する確からしさの知識は PME に与える。CME は処理過程が矛盾に陥らないことを保証する。一方、PME は確からしさに関する情報を利用して、処理を確からしい方向に誘導する。

3.1 処理の流れ

PSE の処理の流れは次のようになる。

[step 1] 前処理

最初の推論環境 $E_0 = \{ \}$ を作って、これを環境の木のルート節点にする。また、 E_0 の確信度を 1 に設定する。

[step 2] 推論環境の選択

PME を呼び出し、環境の木から確からしい環境 E を選出する。環境の木が空である場合、処理を失敗として終了する。

[step 3] 解析処理

環境 E において、次の事象のいずれかが発生するまで、各解析レベルの処理を繰り返して行う。

(A) 入力文に対する意味解釈が生成される。

この場合、PSE は生成された意味解釈を出力し、それを文脈情報として記録した後、処理を終了する。

(B) 曖昧性が生じる。

この場合、PSE は CME を呼び出して、新しい仮定を生成し、環境の木を拡張する。また、各仮定の確からしさを決定する。

(C) 矛盾が検出される。

この場合、CME を呼び出して、矛盾環境を削除する。また、PME を呼び出して、確からしさの修正を行う。

[step 4]

[step 2] へ戻って、処理を続ける。

3.2 仮定の生成と利用

統合パーサを用いる自然言語処理過程は仮定に基づく推論過程である。仮定はその推論過程における役割によって、次のように分類できる。

(1) 直接信念をサポートする仮定

この種の仮定はシステムがどの仮定を信じるかを示す。システムはこのような仮定によって、排他的な推論結果がそれぞれ異なった環境で成り立つようにする。例えば、現在の文脈で文字列「かく」に対する単語 word-1 が動詞「書く」と名詞「角」の両方とも可能である場合、システムはこの曖昧性に応じて仮定 A_1 (動詞と仮定する) と A_2 (名詞と仮定する) を生成し、次の理由付けを CME に与える。

word-1, $A_1 \rightarrow \text{verb-1}$

word-1, $A_2 \rightarrow \text{noun-1}$

例えば次の推論が仮定 A_1 を含んだ環境で続けられるならば、noun-1 がこの環境で成立しないから、システムは verb-1 を word-1 の“論理的な結論”とみなす。

(2) 解析過程を制御するための仮定

この種の仮定はシステムが次に何をすべきかを示す。「小さい小学校の門」という文の構文構造を考えよう。図 5 に示すように、名詞「小学校」に対して、異なった構文規則を利用することによって、二つの解析木が得られる。このような構文的な曖昧性に対処するために、本システムでは解析過程を制御するための仮定を利用する。例えば、 G_1 と G_2 が次の構文規則とすると、

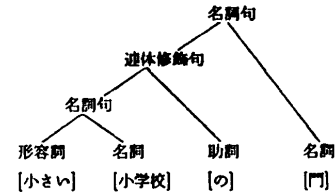
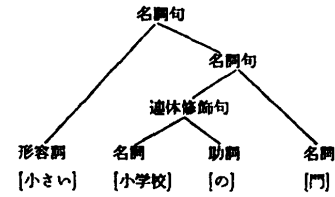


図 5 構文構造の曖昧性

Fig. 5 Ambiguity in the syntax analysis.

G_1 : 連体修飾句 = 名詞 + 助詞 [の]

G_2 : 名詞句 = 形容詞 + 名詞

システムは名詞「小学校」に対して構文規則 G_1 と G_2 のどちらを適用するかを制御するために、仮定 $@G_1$ と仮定 $@G_2$ を生成する。これを利用して、処理の方向を制御する。すなわち、仮定 $@G_1$ を含む環境において、構文規則 G_1 を適用し、連体修飾句「小学校の」を生成し、仮定 $@G_2$ を含む環境において、名詞句「小さい小学校」を生成するようになる。

3.3 仮定の切り換えによる解釈の再構成

現在の環境が矛盾したり、現在の処理結果がある理由によって拒否されて、新しい処理に切り換える際に、これまでの処理結果をできる限り再利用した方がよい。統合パーサによる自然言語解析システムにおいては、解析過程で作られた信念のすべてが仮定の組合せに依存するため、仮定の切り換えによって、新しい環境における解釈を再構成することは容易である。これを示すために、次の例を見よう。

例 1 台風ではしが折れた。 (S1)

ここで、文 (S1) だけを考えると、単語「はし」の同音多義の曖昧性のため、次のように

「台風で橋が折れた」 (I1)

「台風で端が折れた」 (I2)

2通りの解釈が可能である。いま、システムは既に解釈 (I1) を出力したが、ある理由によって (例えば、後続文が「木の枝が短いね」である場合)、解釈 (I1) が拒否されたとしよう。この場合、システムは文 (S1) に対して解釈 (I2) を見つけなければならない。しかし、(I2) と (I1) は構文構造も意味構造も同じであるから、前の処理結果を利用して、仮定を切り換え

ることによって、文(S1)を再解釈することができる。

これを実現するための方法として、変数節点という特殊な節点を利用する。変数節点とはその値(節点で表す信念)が環境に依存するような節点である。環境が変わると、変数節点で表す信念も変わる。

図6は文(S1)の解析過程で作られた信念の間の依存関係を示す。ここで、変数節点 conc-x は仮定 @conc-1 を含む環境において、「橋」と解釈され、また仮定 @conc-2 を含む環境において、「端」と解釈される。したがって、システムは解釈(I1)を生成すると同時に、解釈(I2)も別の環境で生成することになる。

3.4 確からしさの設定

解析の過程で、曖昧性に応じて仮定が生成されると、PSEは知識と今までの処理結果を利用して、各仮定に確からしさを与える。本研究では、当面次のような基準で、仮定の確からしさを設定した。

(1) 単語の語境界の曖昧性に対して、単語の長さによって確からしさを設定する。日本語の場合、長い単語が確からしいので、長い単語候補に高い確からしさに与える。

(2) 単語の同音異義の曖昧性と単語の品詞の曖昧性に対して、各品詞の使用頻度(この頻度は知識として辞書に登録する)と、解析の部分結果による構文的予測によって、確からしさを設定する。使用頻度が高くかつ現在の解析結果から期待されるものに高い確からしさを与える。

(3) 単語の意味概念の曖昧性に対して、対象世界に関する記述や先行文脈情報を参照して、そして、意

味概念の関連関係によって、確からしさを設定する。このために、概念の間の連想関係を表す簡単なネットワークを用いている。

(4) 構文構造の曖昧性に対して、構文構造を構成する単語の数と構文構造に関する知識によって確からしさを設定する。

ただし、確からしさの設定はこの研究の目的ではない。ここで述べた方法はまだ十分ではなく、今後実験的に検討していくつもりである。

4. 実験と評価

我々は統合パーサをベースにして、図1に示すような自然言語解析システムを試作した。システムはSUN Workstationの上でCommonLispでインプリメントした。プログラム全体は約4,000ステップからなっている。そのうち、CMEとPMEが併せて約1,200ステップ、単語辞書や意味辞書の管理、参照などに関する部分が約800ステップ、問題解決エンジンPSE(各解析レベルの処理を行う操作を含む)が約2,000ステップである。

このシステムへの入力ローマ字列で表す仮名文字列からなる日本語の文であり、出力は入力文の意味表現のリストである。システムの知識ベースには言語的な知識(辞書、構文規則など)と対象世界に関する知識(単語の意味概念や意味概念の間の関連など)のほか、文脈情報も格納される。

我々はこのシステムに表1に示す文法規則を与え、約80個の日本語の文を対象として、実験を行った。実験によって、本論文で提案した統合パーサという方法は次のような曖昧性に対して有効であることが確認された。

(1) 単語の語境界の曖昧性

本システムでは、単語の語境界の曖昧性を以下の方

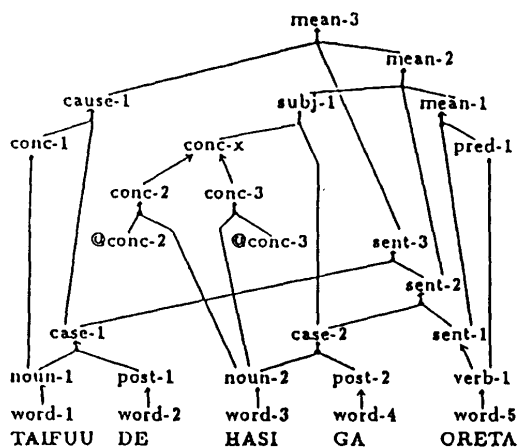


図6 文(S1)の解析過程で作られた言明の間の依存関係
Fig. 6 The dependencies of beliefs produced in analyzing sentence (S1).

表1 構文規則表
Table 1 The grammar.

名詞(句) = 形容詞 + 名詞(句)	(1)
名詞(句) = 連体修飾句 + 名詞(句)	(2)
名詞(句) = 文 + 名詞(句)	(3)
連体修飾句 = 名詞(句) + 助詞(の)	(4)
格 = 名詞(句) + 助詞	(5)
文 = 動詞	(6)
文 = 形容詞	(7)
文 = 名詞(句) + 助動詞	(8)
文 = 格 + 文	(9)
文 = 文 + 終助詞	(10)
文 = 副詞 + 文	(11)

法を統合することによって解消する。

- a) 構文的に期待しない単語の排除
- b) 単語の長さによる確からしさの設定
- c) 意味構造による不適当な候補の排除

例2 「ホテルはえきのむこうである」 (S2)

この文の語境界の曖昧性として、「は、はえ」、「の、のむ」、「で、である」などが挙げられる。しかし、単語「はえ」、「のむ」は構文的な知識によって、初めから排除できる。また、単語「である」が単語「で」より長いから、システムはまずこれを採用する。

例3 「彼がくるまで皆を待つ」 (S3)

「彼がくるまで皆が待つ」 (S4)

この例では、文字列「くるまで」は「車で」とも「来るまで」とも解釈できる。(S3)に対して、長い単語「くるま」を優先的に採用することによって解決可能であるが、(S4)に対して、これは間違いである。しかし、この間違いは文の意味構造の知識によって解消できる。ここで、単語「くるま」をサポートする仮定を A_1 とし、単語「くる」をサポートする仮定を A_2 とし、また、仮定 A_1, A_2 によって生成された環境を E_1, E_2 としよう。システムは最初に環境 E_1 を選択する。この環境において、「彼が車で皆が待つ」という文が得られるが、その意味構造が不整合であるため、システムは環境 E_1 を矛盾環境として削除し、推論を環境 E_2 に移って続ける。この環境において、正しい解釈が発見できる。また、「皆が待つ」という部分に関する処理は既に E_1 で行われたため、その結果は直接に利用できる。

(2) 単語の同音異義の曖昧性

本研究では、同音異義の単語同士を一つの記号として辞書の中に登録し、また、各単語の使用頻度も記録する。処理の過程で単語の同音異義の曖昧性が生じると、システムは各々の可能性に対して、その使用頻度と文脈情報によって、確からしさを決定する。特に、同音異義となる単語同士が品詞としても同じならば、重複した構文解析処理を回避することができる。

例4 「あめがすきだ」 (S5)

この例では、単語「あめ」は「雨」と「飴」の二つの可能性がある。この違いは概念レベルで生じているので、概念レベルでは仮定 @rain と @candy を生成することによって区別する。構文レベルでは仮定は生成されない。いま、システムは仮定 @rain を含む環境でこの文に対する構文解析を完成したとしよう。この場合、仮定 @candy を含む環境にも同じ構文構造

が構成されている。したがって、もし解釈「雨が好きだ」が拒否されるならば、システムは仮定 @candy を含む環境で再び構文解析を行う必要がなく、前の構文解析木を利用して、解釈「飴が好きだ」を生成することができる。

(3) 単語の意味表現の不確定性

単語の意味概念の不確定性に対して、本システムでは、対象世界に関する記述と文脈情報を利用して、可能な意味概念の確からしさを決定し、最も確からしい解釈を出力する。まず、次の例をみよう。

例5 「かぎをかしてください」 (S6)

ここで、対象世界には、「かぎ」は三つがあって、それぞれ図書室のかぎ (key1)、コピー室のかぎ (key2)、そして会議室のかぎ (key3) であるとしよう。この文には「かぎ」の指示対象が言明されていないため、解釈を唯一にしばることができない。

いま、システムに既に次のような知識を対象世界に関する記述として与えられたとしよう。

[対象世界において、「かぎ」で key1 を指す可能性が 0.2、key2 を指す可能性が 0.3、key3 を指す可能性が 0.5 である.]

先行文脈がない場合、システムはこのような知識を利用して、「かぎ」を key3 と解釈する。

一方、先行文脈があった場合、システムは先行文脈に現れた意味概念と「かぎ」の意味概念の間の関連関係によって、現在の文脈で最も確からしい解釈を出力することができる。例えば、先行文脈として、

「資料をコピーしたいのですが」 (S7)

というような発話があった場合、システムは「かぎ」を key2 と解釈することができ、また、次のような発話

「本を捜したいのですが」 (S8)

があった場合、システムは「かぎ」を key1 と解釈することができる。

この処理では、概念間の連想関係を記述するため、D. Waltz らのマクロ素性³⁾の手法と類似の関連ネットワーク (図 7) をあらかじめシステムに与えておく。このネットワークの節点は意味概念あるいは事象を表

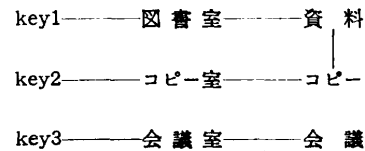


図 7 意味概念の関連ネットワーク
Fig. 7 An Associative Networks between concepts.

し、節点の間の弧はその両端の意味概念あるいは事象に関連していることを意味する。システムはこのネットワークを利用して（対象世界に関する記述も参照する）、「かぎ」の可能な意味概念の確からしさを決定する。

図8には先行文脈が(S7)である場合、文(S6)に対する解析過程で作られた仮定と環境の木を示す。

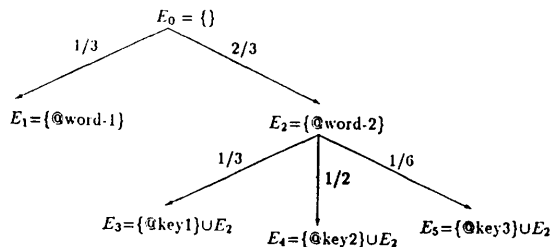
この方法は単語自身の多義性によって生じる曖昧性にも適用できる。次の例をみよう。

例6 「はなしてはいけない」 (S9)

この文において、単語「はなす」は「話す」と「離す」との二つの可能性がある。もし先行文脈として、「静かにしなさい」が与えられているとき、システムは単語「はなす」を「話す」として解釈する。また、先行文脈として、「つりわにしっかりつかまっていなさい」が与えられているときは、システムは単語「はなす」を「離す」と解釈する。ここで、システムは動詞「はなす」の二つの可能な概念に対して、それらが先行文脈に現れた単語（例えば、「静か」、「つかまる」）の概念との関連関係によって、どれを採用するかを決定する。

(4) 構文構造の曖昧性

構文構造の曖昧性に対して、本システムは形容詞の修飾関係や動詞に対する格関係などの知識を利用して、確からしさを設定し、最も確からしい構造を選択する。例えば、形容詞はそれに最も近い名詞を修飾する可能性が高いから、本システムは前述の「小さい小学校の門」という文字列に対して構文解析を行う際に、まず形容詞「小さい」を名詞「小学校」と結合する。



仮定の意味

- @word-1: 最初の単語が「か」である。
- @word-2: 最初の単語が「かぎ」である。
- @key1: 単語「かぎ」が key1 を指示している。
- @key2: 単語「かぎ」が key2 を指示している。
- @key3: 単語「かぎ」が key3 を指示している。

図8 文(S6)の解析過程で生成された環境の木
Fig. 8 The Environment Tree constructed in analyzing sentence (S6).

もう一つの例として、次の二つの文をみよう。

例7 「彼が病気で入院したことを聞いた」 (S10)

「彼が病気で入院した友達を見舞った」 (S11)

これらの文は品詞列として全く同じであるが、その構文構造は異なる。しかし、文(S10)において、単語「こと」が動詞「入院した」の主語にならないから、システムは表1の構文規則(9)を採用し、「彼が病気で入院した」という部分文を生成する。また、文(S11)に対して、文法規則(3)を採用し、「病気で入院した友達」という名詞句を生成する。

5. 関連研究との比較

本研究と関連する研究として、Mellishの早期意味解析⁶⁾とWaltz and Pollackのコネクショニストアプローチ³⁾とHirstのポラロイド語⁷⁾などがあげられる。

本研究と上に列挙している研究との共通点は、自然言語処理過程における各解析レベルの処理を並行して行うことによって、自然言語の曖昧性を早期解消することである。本研究では、自然言語処理過程を仮定に基づいた推論過程の一種として取り扱い、それに関して必要となる論理的な整合性維持の機能と確からしさによる推論誘導の機能を融合したメカニズムを提供した。

Mellishの早期意味解析とWaltz and Pollackのコネクショニストアプローチと同様に、統合パーサは、知識と解析の部分結合を統合することによって早期に曖昧性を解消することができる。また、確からしい解釈を見つけることもできる。しかし、彼らの方法では、解析が誤った方向に進んだとき、矛盾検出後のbacktrackingが高価になる。また、矛盾が生じた場合における仮定の確からしさの修正方法は提供されていない。我々の統合パーサでは、すべての処理方向がCMEによって保持されているから、見つけた解釈が否定されても、仮定の組み合わせの切り替えによって、次の解釈を容易に見つけることができる。また、処理の進行に伴って、仮定に関する確からしさ情報を動的に修正することによって、システムの処理を常に確からしい方向に進めることが保証できる。

一方、Hirstのポラロイド語は、文中に現れる単語に対して、初めにすべての可能な意味概念を与え、処理の進行に伴って、次第に整合性のない意味概念を排

除することによって、解釈を絞り込んでいくという方法をとっている。しかし、この方法では、日本語対話文のような曖昧性の多い場合、解析の初期に膨大な数の候補が作られる。これに対して、我々の統合パーサは、処理の過程で最も確からしいものだけを取り出して処理を進めるといった解析方法をとったので、確からしさの低い候補による不必要な処理を避けることができる。

6. おわりに

本論文では、統合的自然言語処理を効率的に実現するための新しいメカニズムを提案した。この方法では、統合的自然言語解析システムは統合パーサに基づいて構築される。統合パーサは整合性維持エンジン CME と確からしさ管理エンジン PME と二つのサブシステムからなる。CME は統合的処理の過程で生成された言明の集合の論理的な整合性を維持し、可能な推論方向を木の形で保持する。一方、PME は仮定に与えられた確からしさ情報を信念ネットワークの形で動的に管理し、処理を確からしい方向に誘導する。

我々はこの統合パーサを用いて、統合的自然言語解析システムを試作し、実験を行った。このシステムは解析の過程において、各解析レベルの知識や解析の部分結果を統合して利用し、曖昧性をできるだけ早期に解決する一方、本当に曖昧な文に対しても、最適な解釈を出力することができる。

今後の課題としては、複雑な文を解析できるように、今まで試作したシステムを拡張することがあげられる。また、確からしさの設定も今後の課題の一つとなる。さらに、本研究で実現した統合パーサの基本的な考え方を人工知能の他の領域、例えば音声認識、文字認識などのような誤りを含んだ入力の場合に適用して、検討してみたい。

謝辞 日頃、ご討論いただく京都大学工学部堂下研究室の諸氏に深く感謝いたします。

参考文献

- 1) de Kleer, J.: An Assumption-Based TMS, *Artif. Intell.*, Vol. 28, pp. 127-162 (1986).
- 2) Pearl, J.: Distributed Revision of Composite Beliefs, *Artif. Intell.*, Vol. 33, pp. 173-215 (1987).
- 3) Waltz, D. and Pollack, J.B.: Massively Parallel Parsing: A Strongly Interactive Model of Natural Language Interpretation, *Cognitive Science*, Vol. 9, pp. 51-74 (1985).
- 4) Liu, X., Nishida, T. and Doshita, S.: Maintaining Consistency and Plausibility in Integrated

Natural Language Understanding, *Proceedings of ICSC-88 (International Computer Science Conference '88)*, pp. 360-367 (1988).

- 5) 劉, 西田, 堂下: ATMS の拡張とその統合的自然言語理解システムへの応用, 第 36 回情報処理学会全国大会論文集, 4 T-1 (1988).
- 6) Mellish, C. S. (田中穂積訳): 自然言語意味理解の基礎, サイエンス社, 東京 (1987).
- 7) Hirst, G. and Charniak, E.: Word Sense and Case Slot Disambiguation, *Proceedings of AAAI-82*, pp. 95-98 (1982).

(平成元年 7 月 7 日受付)

(平成 2 年 6 月 4 日採録)



劉 学敏 (正会員)

1956 年生。1984 年中国北京大学計算機科学技術系卒業。1988 年京都大学大学院工学研究科情報工学専攻修士課程修了。現在、同大学院博士課程在学中。仮説推論、自然言語理解の研究に従事。



西田 豊明 (正会員)

1954 年生。1977 年京都大学工学部情報工学科卒業。1979 年同大学院修士課程修了。1980 年同大学院博士課程退学。同年より、京都大学工学部助手。1988 年 6 月助教授。人工知能基礎、特に定性推論と空間推論、自然言語理解などの研究に従事。京都大学工学博士。1984 年から 1 年間 Yale 大学客員研究員。1988, 89 年人工知能学会全国大会優秀論文賞。1988 年度人工知能学会論文賞。情報処理学会 30 周年記念論文賞。著書「自然言語処理入門」(オーム社) など。人工知能学会、認知科学会、日本ソフトウェア科学会、電子情報通信学会、AAAI, ACL 各会員。人工知能学会編集委員など。



堂下 修司 (正会員)

昭和 33 年京都大学工学部電子工学科卒業。昭和 35 年同大学院工学研究科修士課程電気工学専攻修了。昭和 38 年同博士課程単位取得退学。同年京都大学工学部助手。昭和 40 年同助教授。昭和 43 年東京工業大学助教授。昭和 48 年京都大学工学部教授。その間、音声の分析と認識、オートマトンの学習の構成、自然言語処理、人工知能など知的情報処理の研究に従事。京都大学工学博士。昭和 34 年度通信学会稲田賞受賞。昭和 63 年度人工知能学会論文賞受賞。情報処理学会 30 周年記念論文賞。人工知能学会、電子情報通信学会、音響学会等各会員。