

# 簡単な日本語の意味と文法を学習するシステム

高木 朗† 戸沢 実†† 太田善之†† 篤田 洋†† 伊東幸宏†† 小原啓義††

†† 関CSK総合研究所

†† 早稲田大学理工学部

## 1. はじめに

人間がなぜ言語を獲得しうるかという問題は、心理学・言語学における大きな問題である。人工知能においても、言語を学習する能力を実現することは、システムに予め完全な文解釈・生成能力を与えておかななくてもすむという意味で、重要なテーマである。この為、これまでも言語学習能力をコンピュータ上で実現しようとする研究がいくつか見られる[1][2]が、いずれも単純なもので、単語の意味と文法の両方を学習するシステムは検討されていない。本稿では、視覚入力と言語入力の二系統の入力を備え、視覚情報を手掛りにしつつ、単語の意味と文法を学習していくシステムについて述べる。

## 2. 基本的な考え方

言語獲得とは、表層文における単語の意味や構文構造に関する知識を全く持たない状態から、それらの知識を獲得していくプロセスである。従って、表層文だけを手掛りに、そこから単語の意味や構文構造を学習することは出来ない。言語とは独立した情報が表層文と対にして与えられる必要がある。幼児の言語獲得においては、視覚情報が大きな役割を担っていると考えられている[3]。幼児の言語獲得のプロセスは十分明らかにされているとは言えないが、恐らく幼児は、親が与えた文が目の前で起こっている現象を言及するものであると判断し、その視覚現象の認識に基づいて与えられた文の意味を推定すると考えられる。これにより、文中の未学習の単語の意味を学習し、更に文中で与えられた単語の意味とその並びを、推定した文の意味と比較することにより、単語の並べ方と句・節・文の意味との間の関係、即ち文法に関する知識を獲得していくものと考えられる。

本システムでも、この考え方に基づいて言語学習を行っている。これを可能にする為には、言語情報と視覚情報が密接につながっている必要がある。即ち、視覚現象に対する認識と、言語を解釈して得られる意味とが共通のデータ構造を持つか、少なくとも互いに交換可能な構造を持つということである。本稿では、

- (1) 人間は、目を見たことを言葉で表現できるし、逆に言葉で言われたことを目で確かめるという操作を行うことができる。
- (2) 視覚に限らず、味覚・触覚・痛覚等についても言

えることであるが、感覚刺激に対する認識の結果は、区別なく次の認識・思考に利用でき、等しく言語化の対象になる。

- (3) 言語についても、それを理解するという事は、言語情報を“認識する”ということであり、やはり次の認識・思考のデータとなりうる。

等の理由から、視覚と言語が共通のデータ構造によって結ばれていると仮定し、視覚と言語に共通な“深層構造”を想定している。

この様な深層構造を考える場合、視覚情報が言語の句構造をベースにしては記述できないこと、及び形態素の並び順序というものが視覚情報においては意味を持たないこと等の理由により、従来の変形文法における句構造標識的な構造は採用できない。この為、我々は、この深層構造が満たすべき条件として、

- (1) 視覚情報の特徴解析結果を用いて、その視覚現象を記述する能力があること。
  - (2) 深層構造から、これを言語に変換する際に必要な単語と品詞及びそれらの単語間の意味接続関係がわかること。
  - (3) 同義文に対する深層構造が同一の構造を持ち、同一の視覚現象と対応付くこと。逆に、一つの特徴解析結果から同義文を生成する能力を持つこと(同義文の生成能力がないと、視覚情報は決まりきった表層文としか対応が付かず、視覚と言語を接続したとは言いがたい)。
  - (4) 単語(形態素)分割を前提とした構造をとらないこと(学習の初期の段階では、学習者は単語知識を持たない為、単語及び句構造分割をベースとしたデータ形式は不相当である)。
- 等を取り上げ、生成意味論的な立場からその表現手法の検討を行ってきた[5][6][7]。

この深層構造(即ち意味表現)を採用した場合、

- (1) 単語の意味とは、深層構造(以下、意味表現)の断片である。
- (2) 文の解釈・生成とは、表層文と意味表現との間の相互変換である。
- (3) 従って、この時の文法とは、この相互変換に必要な知識であり、単語がどのように並んでいる時に、単語の意味である意味表現をどのように接続したら、句・節・文としての意味表現が構成されるかの知識である。

という特徴がある。本システムでは(1)と(3)の学習を行う。単語の意味は、基本的には視覚情報から生成され

る意味表現を写し取ることで学習される(格助詞等、直接視覚情報中に対応する物がない語でも、視覚情報中から命題の意味表現が生成されるので、そこから対応する部分を切り取ることで学習される)。文法学習は、「表層文」と、「表層文の意味として視覚情報から推定された意味表現を表層文中の単語で分割した、単語間の意味接続を表す木構造」とを対にして学習される。これは、基本的には一次元ストリングと木構造との間の変換規則の学習としてとらえられる。

### 3. 意味表現の基本表記

意味表現の記述に用いる記号の定義を以下に示す。

- (a) 名詞を○nodeで表す。
- (b) 助動詞を◎nodeで表し、動詞を◎◎nodeで表す。  
これは、動詞内部に助動詞が含まれていると考えているためである。なお、◎◎を単に◎のみで略記する場合もある。
- (c) 助詞、前置詞を介、↑等のarcで表す。ただし、介は主格を、又、↑は目的格を表す。他の格を表す表現は必要に応じて定義する。
- (d) 従属節を( )で囲み、主節と区別する。
- (e) 関係代名詞を●nodeで表す。また、関係代名詞がどの名詞を受けているかを表すために、先行詞と●とを=で結んで対応関係を表す。これにより、先行詞及び関係節は

○≠●+文、動詞句)

という形で表される。

- (f) 助詞「の」(of)の中にあって、それに接続する名詞句を受ける代名詞相当のnodeを\*nodeで表す。
- (g) to不定詞中であって、それに接続する動詞句を受ける代動詞相当のnodeを⊗nodeで表す。

### 4. システムの構成

図1に本システムの概略を示す。

このシステムは図形入力と、未知語もしくは未知文型を含んだ言語入力を受け取る。この二種類の入力は

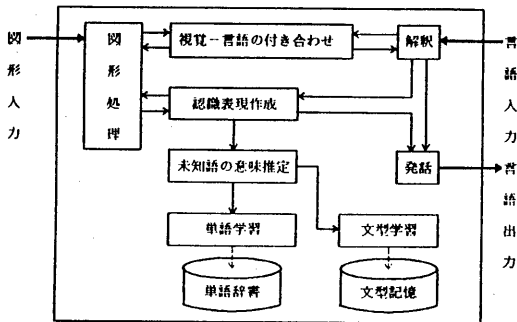


図1 システムの概略

対にして与えられ、システムは図形情報を参照しつつ、入力文に対応する意味表現を推定する。次いで、推定された意味表現から、入力文中の既知語の意味表現を差し引いて未知語の意味を推定し、これを未知語と連合させて単語辞書に登録する。又、新入力文を解釈・発話する為に必要な文法知識を学習し、文型記憶に登録する。

#### 4.1 図形入力及び図形処理

本システムに与えられる図形入力は1枚或いは複数の画面である。複数の画面はそれを一つの光景の連続した描写であるとみなすことによって動画を表すものとする。それぞれの画面は130×200のドット・マトリクスで構成され、画面中の図形は線分又は円弧からなる単純な二次元図形である。これらの図形は図形処理部で特徴解析され、その結果が意味表現の形式に変換される。画面から生成可能な認識(意味表現)のタイプを表1に示す。なお本システムでは、認識のタイプ及び主語・主動詞を指定すれば、図形情報からその画面に対応した一つの認識を表す意味表現を得ることが出来るようになっている。

#### 4.2 言語入力及び解釈

本システムに与えられる言語入力は、画面中の指示された図形の存在の有無、或いは図形の持つ特徴について言及したもの、図形の持つ属性の変化等について言及した日本語の平叙文及び名詞句である。取り扱う語句の範囲を表2に示す。これは、学習可能な文構造の範囲でもある。入力文はローマ字で分かち書きされている。しかし、システムは始めから単語分割を知っている訳ではなく、未知語は、空白を間に挟んで複数単語にまたがってもよいものとしている。未知語の数は原則として文中に一語のみを許す。

言語入力は、既に習得した文型記憶を用いて、単語の係り受けを表す木に変換される(図2(b))。この木は必要に応じてprimitive levelの意味表現に展開される(図2(c))。しかし、

- 入力文中に未知語が存在する。
- 入力文を正しく解釈する為の文型を知らない、或いは文型が不完全である(更に新しい句が付きうる)。等により“木”が一つにまとまらなかった場合には、係り受けの明らかな部分だけをまとめた“部分木”を作る。この際、名詞・動詞の修飾句(節)の正しさを

表1 画面から生成可能な認識のタイプ

画面	タイプ
静止画	着目した図形が属する概念のクラスについて 着目した図形の存在、又は位置について 着目した図形の図形的特徴(色、大きさ、形等について) 着目した図形がもつ部分図形について
動画	着目した図形の属性値変化について

確かめる為に画面情報と付き合わせ処理【5】を行う。この時、学習の過程では、次のような問題が生じる。

例えば、初めて「リンゴ」という名詞の意味を画面と対応付けて学習した時は、「リンゴ」とは「ある特定の色・形・大きさ及び部分を持ったもの」というように記憶される。この為、次に大きさが異なる「リンゴ」を画面中に与えた場合に単語辞書中の「リンゴ」と対応付けることができない。言語入力画面中の図形について言及したものであるから、このような場合は「リンゴ」の単語辞書の意味表現と、画面中の同じ形・部分を持つ図形との対応を仮定して、異なる属性値

表2 取り扱う語句の範囲

[A] 名詞	
a. 図形を表す名詞(以下、図形名詞)	「ロボット」「木」等
b. 部分図形を表す名詞(以下、部分名詞)	「コップの <u>と</u> って」等
c. 属性を表す名詞(以下、属性名詞)	「色」「数」「位置」「速さ」等
d. 数詞	「1本」「3つ」等
e. 代名詞	「これ」「これら」
[B] 動詞	
a. 属性値の時間的変化を表す自動詞	「動く」「転がる」等
b. 「もっている」「だ」(「である」)「ある」「等しい」	
c. 形容詞、形容動詞 終止形(原級、比較級、最上級)	「赤い」「大きい」「～より大きい」「～の中で一番大きい」等
[C] 連体修飾表現	
a. 属性値を言及するタイプ	
a.1. 形容詞連体形	
i) 値を直接言及するもの	「赤い」「丸い」等
ii) 標準値との比較結果を言及するもの	「大きい」「広い」等
a.2. 属性値を言及するタイプの形容詞に類した表現	
i) 「～の」	「10の面積の」等
ii) 所有格関係節	「面積が10の」等
iii) 主格関係節	「10の面積をもつ」等
b. どのような部分をもっているかを言及するタイプ	
i) 主格関係節	「長い足をもつテーブル」等
ii) 所有格関係節	「足が長いテーブル」等
c. 数を言及するタイプ	「3つの」「5本の」等
d. 部分名詞を修飾するタイプで、その部分名詞がどの図形の部分であるかを言及するもの	「このコップの <u>と</u> って」等
e. 属性名詞を修飾するタイプで、どの図形の属性が問題となっているかを言及するもの	「ボールの色」等
f. 指示を表すタイプ	「この」「これらの」
[D] 連用修飾表現	
a. 属性値を言及するタイプ	
a.1. 属性形容詞連用形	「速く」等
a.2. 副詞	「ゆっくり」等
a.3. 副詞に類した表現	
「～の速さで」「～に向かって」等	
b. 位置を言及するタイプ	「机の上を」「家の前を」等
c. 同時に起っている他の属性値変化を言及するタイプ	「転がりながら」「膨らみながら」等

を持つ「大きさ」を表す意味表現の枝を消去し、代わりに「リンゴ」の大きさに関する標準値を記憶する。標準値は、「大きい」・「小さい」等の比較を表す認識を作成する為に用いられる。動詞についても同様の処理が行われる。形容詞の連体形についても同様な抽象が行われるが、これは、後の未知語の意味推定部で行う。

#### 4.3 入力文の意味推定

入力文中の未知語に対応する意味表現を推定し、その語を解釈・発話する為に必要な文型を獲得する為には、まず入力文全体の意味を推定しなければならない。ここでは、図形処理の結果を手掛りにしつつ入力文の意味表現を推定する。

##### 4.3.1 解釈結果のまとめあげ

本システムでは、前述のように図形情報から生成すべき認識のタイプ、主語及び主動詞をパラメータとして与えることにより、入力画面に対応した表2の範囲内の任意の意味表現を、得ることができる。さて、入力文の意味を推定する為には、解釈結果の部分木の意味からこのパラメータを抽出する必要がある。しかし、入力文中でパラメータの推定に必要な部分が未知語であったり、複数の部分木に分散しているような場合には、部分木を一つの木にまとめてみなければパラメータを抽出出来ない。部分木を一つにまとめる為には、各部分木が他の部分木とどのように接続するのかわかる必要があり、その為に各部分木中の構文的・意味的情報を抽出する必要がある。この為、ここでは認識フレームと呼ぶものを用意している。

##### 4.3.2 認識フレーム

例文: 「青いコップは机の上の赤いコップより大きい。」

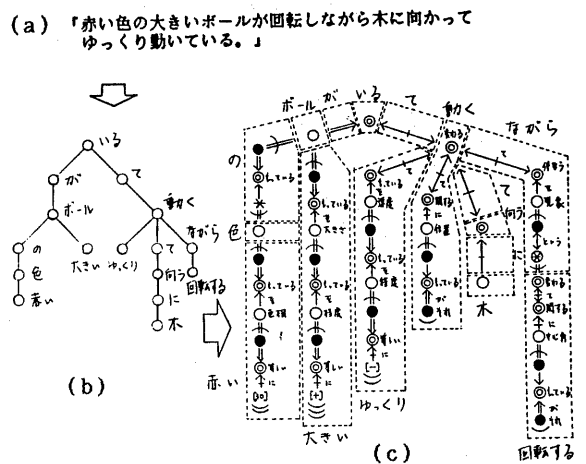


図2 解釈例

この例文は、

- (1) 「コップ1は青い」
- (2) 「机の上にコップ2がある」
- (3) 「コップ2は赤い」
- (4) 「コップ1はコップ2より大きい」

という主張を含んでいる。即ち、1つの文の主張する内容は幾つかの単文に分解することが可能である。このような単文を認識の単位と呼ぶ。この認識の単位は、表1の認識のタイプに対応する。これらの認識の単位内では意味表現の構造は一定である。そこで、この認識の単位ごとにフレームを用意し、各部分木をたどりながらそれらの意味が、どのフレームに対応するか

(単数とは限らない)を推定する(表4に認識フレームのスロットの構成を示す)。この為、各フレーム毎に、それに対応する意味表現の骨格を表すskeletonを用意する。skeletonは認識の単位内の意味表現の構造を、従属節に関する構文情報を除いて記述し、更に名詞成分(図形・部分・属性・値)、一般動詞成分(変化を表す動詞)が位置する場所をスロット名(表4のsslot\*)に置き換えたテンプレートである。表3にフレームの種類を、図3にskeletonの例を、図4にフレームの例を示す。

システムはskeletonを部分木の意味表現に当てはめることにより、意味的に対応するフレームを選ぶ。この当てはめの際には、skeletonのsslotに対応した名詞成分・動詞成分(意味表現)をフレームのスロットに記入する。更に、部分木の末端の意味成分等を別のスロットに記入する。これにより、各部分木がどのような語に係り得るか、どのような語に係り得るかを推定することができる。

#### 4. 3. 3 部分木のまとめ

上述のようにして、フレームによって集めた情報を手掛りにして部分木を一つの木にまとめようとする。まず、フレームに記入された、部分木の末端に関する

情報を付き合わせ、これにより、二つの部分木をつなぐ語が推定される。次に、末端情報を含むフレームのスロットの状態から、部分木同志の意味的な接続可能性を調べ、その二つの部分木を一つの木にまとめる。この時、属性名が未知語の場合や、関係節の学習の場合に意味的情報と構文的情報の間で矛盾が生じる場合があるので、その解決方法を予めフレームごとに記入しておく。部分木が多数存在する場合は、二つずつまとめるとめる。但し、図形情報に対する認識を指定するパラメータが揃ったならば、それ以上部分木をまとめる必要がないので、処理を終了する。

入力文がすべて未知語である場合など、必要なパラメータが揃わない場合や、部分木をまとめる為の情報不足している場合は、システムに予め用意した“構え”(幼児が備えていると思われる外界事象に関する先天的な同定能力に対応するもの)によって指定する(表5)。

表4 認識フレームの内容

スロット名	記入してある、又は記入される内容	
frame	フレームのタイプ名	
skeleton	認識の単位の意味表現の骨格	
sslot*	s1	skeletonの中で指定されるスロットである。部分木の対応する意味表現成分と表層語を対して記入することによりskeletonのどこが部分木に対応するかが分かる。
	s2	
	:	
	sn	
info	sslot	sslotに代入可能な意味成分のリストがsslot毎に与えられている。skeletonを部分木に当てはめる際に用いられる。
	subj	sslotのうち主語になりうるものリスト
flag	subj	部分木の主語に対応するsslotが代入される。
	main	部分木が主語を持つ文である時にflagをたてる。
	upper	部分木の上(部分木に係る方向)の末端の意味表現と表層語及びそれらがskeleton中のどこであるかを同定するためのsslotを連合して記入する。
	lower	部分木の下方に関してupperと同様。但し、部分木の末端とskeletonの末端が一致している時は作成されない。このupper, lowerの情報に基づいて部分木の統合を行う。
proc	asp	部分木から得られる動詞的な相情報
	copula	部分木中でskeletonにないcopulaが挿入されていた場合flagを立てる。
	cause	部分木中でskeletonにないcauseが挿入されていた場合flagを立てる。
proc	proc1	フレームの個別の処理(特に意味的な)を行うプログラムと、それに必要な知識
others	一般動詞を修飾する「伴って」「よって」や、意味表現が特殊な枝分かれをするなどで、複数のフレームがskeletonで扱うことができない状態で接続している場合に対応するためのもの。	
root	部分木によっては、対応するフレームを一意に決定できない。このような場合に備えて他のフレームに対応する可能性を調べるためのもの。	

表3 認識フレーム

例文	フレーム名	
名詞の属性に関するもの	POSS-COLOR POSS-SHAPE POSS-SIZE	赤いコップ ボールは丸い 一番大きいコップ
変化の動詞を表すもの	CHANGE-ATT CHANGE-LOC ROTATE-1	ふくらむ 動く、歩く 回転する、転がる
変化の動詞の属性に関するもの	POSS-VATT	10の速さで(歩く) ゆっくり(動く) 大きく(振る)
図形の存在・位置関係	EXIST-1	コップがある 机の上にあるコップ
部分図形に関するもの	POSS-SUBPART EXIST-2	コップはとってをもつ コップにはとってがある
図形の数	POSS-NUMBER	3つのボール
概念クラス	INC-1	これはコップだ

#### 4. 3. 4 認識作成

以上のようにして得たパラメータと図形処理結果とから意味表現を作成する。これが、入力文全体に対応する意味として推定された意味表現となる。

#### 4. 3. 5 未知語意味推定

入力文中の未知語の意味を推定し、4.3.3 でまとめられなかった部分木を一つの木にまとめる為には、4.3.4 で得た意味表現から、部分木の各単語に対応する意味表現を減算していけばよい。その結果、余った部分が未知語の意味になる。

ところで、多義語の意味は、それが係る語、係られる語によって1つに同定される。このような、多義語の意味の同定に必要な語が未知語であったり、多義語が未だ学習していない意味で使われている場合がある。このような場合には、多義語の各意味の意味表現上の基本構造 (品詞概念に相当) が同一であると仮定して、

s1 ⇒ ◎ ← s2 ⇒ ◎ ← s3 ⇒ ◎ ← s4  
(OBJECT) POSS (COLOR) POSS (HUE) EQ (n-value)  
(SUBPART)

(a) POSS-COLOR

(SIZE) (LENGTH) (DEGREE) EQ (±)  
(SUBPART) (OBJECT) (WIDTH) (n-value) s1 ⇒ ◎ ← s2 ⇒ ◎ ← s4 ⇒ ◎ ← s5  
↓ ↓ ↓ ↓  
POSS POSS POSS EQ FORV  
↑ ↑ ↑ ↑  
s3 (ORDNUM) s8 s6 (STANDARD)  
(n-value) ↓ ↓  
s10 ⇒ ◎ EQ s7 (OBJECT)  
(OBJECT) ↑ ↓  
s9 (n-value)

(b) POSS-SIZE

図3 skeletonの例

```
POSS-COLOR .....
((FRAME . POSS-COLOR)
 (SKELETON . ( S1 => DO POSS (- S2 =>
               DO POSS (- S3 => DO EQ (+ S4 ))
 (SSLOT (S1)
         (S2)
         (S3)
         (S4))
 (INFO (SSLOT (S1 OBJECT SUBPART )
              (S2 COLOR)
              (S3 HUE)
              (S4 N-VALUE))
 (FLAG (SUBJ S1 S2 S3 ))
        (SUBJ )
        (MAIN )
        :
        :
 (PROC (COGTYPE . (LAMBDA (&FRM)
                    :
                    :
                    (SLOTFILLER . (LAMBDA ( .....
 (ROOT .....
 (OTHERS .....
 )
```

図4 認識フレームの例

推定する。

#### 4. 4 文型学習

ここでは、4.3 で推定された入力文の係り受け構造を表す木を基に、入力文を解釈・発話するために必要な文型を学習する。

##### 4. 4. 1 文型

ここでいう文型とは、表層文を解釈して意味表現に変換したり、逆に発話する為に必要な知識であり、換言すれば表層文 (一次元) の語順と、意味表現の接続関係 (二次元) との間の変換に関する知識のことである。

このような知識としては、

- (1) 文中の文構成要素 (単語に相当) が、文に対応する意味表現においてどの部分に対応するかについての記憶
  - (2) その要素が表層でどのような順序に並ぶかについての記憶
  - (3) その要素が文の意味表現上で、他の要素とどのような接続をしているのかについての記憶
- の3つが考えられる。本システムでは(1), (2)をまとめた語順記憶と、(3)に対応する接続記憶をペアにして1つの文型を構成し、これを文型記憶内に蓄積する。

この2つの記憶を用意すれば、解釈において入力文の単語列に一致する語順記憶を探し、それに対応する接続記憶通りに単語を二次元に配列して各単語をそれに連合した意味表現に展開すれば、入力文に対応した意味表現を得ることができる。また逆に、意味表現から上記の逆のプロセスを行えば、文を生成することができる。

又、自然言語において同一の意味内容及び意味的接続関係を持った単語の集合が存在することを踏まえて、各文型は、意味構造及び意味接続関係 (どの様な語に係り得るか) の共通な単語同志をカテゴライズしたも

表5 静止画に対する構え

画面上の物体の数	指示	解釈結果	認識作成
1	なし	未知語のみ 名詞句, 未知語	名詞句 ~がある (EXIST)
	あり	未知語のみ 名詞句, 未知語 形容詞 (連体形) 主格名詞句, 名詞句, 未知語 名詞句, 形容詞 (連体形), 未知語 名詞句, 部分名詞句, 未知語	名詞句 名詞句 形容詞 (終止形) ~は~だ (INCLUDE) (COPULA) 名詞句 or (POSS)
2	なし	主語のない形容詞終止形, 未知語	名詞句 or (COPULA)
	あり	同一概念の名詞句, 形容詞, 未知語 相違概念の名詞句, 「ある」, 未知語	比較の名詞句 or (COPULA) 位置関係 (EXIST)
3以上	あり		最上級の名詞句 or (COPULA)

※ 動画の場合は、画面中の変化に着目して認識を作成する。

の（学習の最終段階においては品詞に相当する）を用いて記述している。

#### 4.4.2 文型学習

ここでは、入力文の意味推定の結果1つにまとまった単語間の係り受けを表す木（この段階では、未知語の意味は既に推定されている）と表層における単語の順序とから、以下に述べる手続きを木のheadから末端へと再帰的に繰り返して文型中の文型記憶を更新し、或いは新しい文型を付加する（図5を参照）。

- (1) 入力文中の各々の語がどのようなカテゴリに分類されるかを調べる。システム中にあるカテゴリの各々と比較し、一致しない場合には新カテゴリを発生させる。
- (2) 文型記憶中の一つの文型に着目して、その接続記憶（Xとする）のhead（句を支配する語、名詞句であれば名詞）と入力文の係り受けの木（Yとする）のheadのカテゴリを比較する。
- (3) (2)において両者のカテゴリが一致するか、或いは意味構造及び意味接続関係を考慮して、同一のカテゴリに抽象できる場合には、Xのheadに更に新しい句が付きうる可能性があるため、XとYの枝同志を比較する。カテゴリが一致あるいは抽象出来なかった時にはYのための新文型を発生させる。
- (4) XとYの木構造の枝同志を比較する際に、Xとペアになっている語順記憶を参照して、枝とheadとの表層における前後関係をチェックした上で入力文の木に新しい枝が存在した場合には、X及びそれとペアになる語順記憶を更新する。

#### 4.5 単語分解

##### 4.5.1 単語分解操作の必要性

単語学習の初期の段階では、本来複数の単語に分解されるべきものが、まとまって一つの“語”として認識されてしまう場合が必ず存在する。本システムにおいても、単語辞書中にある単語が更に複数語に分かれることがわかった場合に、意味表現を分解して新たに単語辞書に登録し直

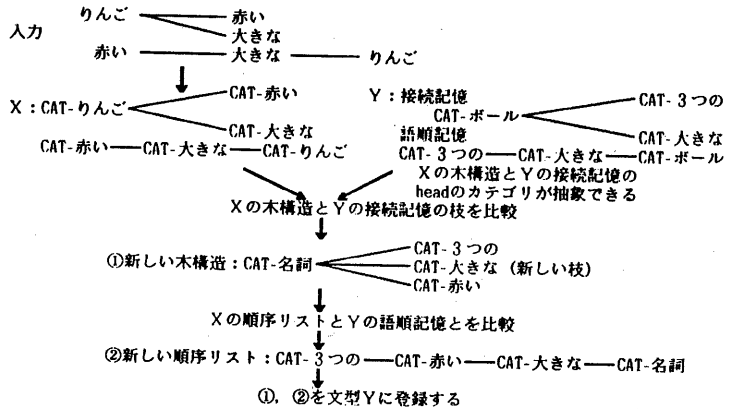


図5 文型更新の様子

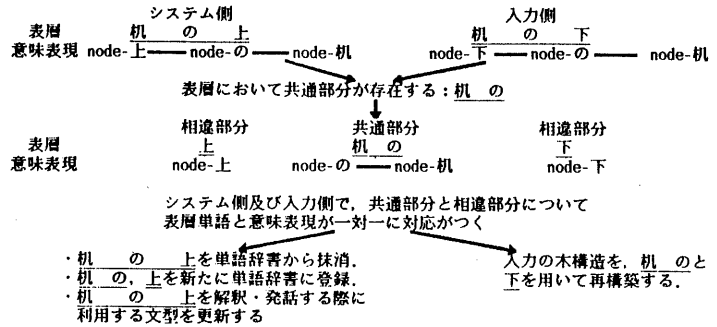


図6 単語分解例 (i)

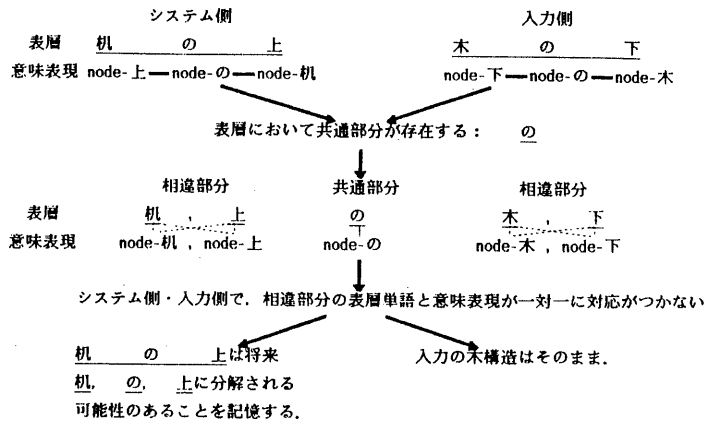


図7 単語分解例 (ii)

し、分解する前の語を解釈・発話する際に使用した文型を更新する。

#### 4. 5. 2 単語分解の手順

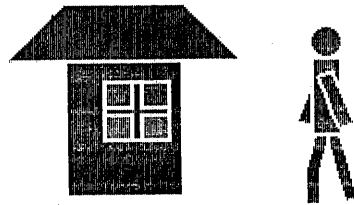
本システムでは以下のようにして単語分解を行っている。

- (1) その時点で単語辞書中に存在する単語の中で、その単語に対応する表層部分が空白で分かち書きされている単語を別に記憶しておく。
- (2) それ以後の入力文中に、(1)によって記憶した単語と、表層において共通する部分が存在することが発見された場合、意味表現上での同一部分を切り出す。
- (3) 入力文中の単語、及び単語辞書に存在する単語で、各々の意味表現の共通部分・相違部分が、表層で一対一に対応がつかずならば、(1)で記憶した単語を単語辞書から抹消する。同時に、分解された複数の単語を単語辞書に登録し直す。
- (4) (3)において登録し直す前の単語の属しているカテゴリがheadである文型を更新する。即ち、その文型のheadの語に係っている各枝(各文型)が、分解された複数単語のうちどの単語に係り得るかを判断し、文型を更新する。

図6、7に単語分解の例を示す。図6は、意味表現と表層の語が一対一に対応付けられた場合を示し、図7は、一対一対応が得られず、その単語が将来どのように分解されるのかを記憶しておく場合を示す。

#### 5. 処理例

図8に単語の意味学習及びそれに伴う文型学習、単語分解の例を示す。入力画面は連続して入力された画面のうち一枚のみを示した。また、図9に学習を進めていくための入力文のシーケンスの例を示す。



(a) 入力画面

```

**** STATE OF SYSTEM (SCE MEMORY) ****
#SCE100004
EL.....(#CAT100006 #SCE100000)
      (GA / IE)
WO.....(#SCE100000 #CAT100006)
#SCE100006
EL.....(#CAT100008 ((#SCE100004) (#SCE100007))) 接続記憶
      (TE I-RU ((GA) (ARU-I)))
WO.....(#SCE100004 #SCE100007 #CAT100008) 語順記憶
#SCE100007
EL.....(#CAT100009)
      (ARU-I)
WO.....(#CAT100009)

```

(b) 新入力前の文型の内容

```

**** INPUT SENTENCE ****
( IE NI HUKA-TSU TE AO-I ROBOT GA ARU-I TE I-RU ) 入力文
**** RESULT OF INTERPRETATION ( TOKEN ) ****
( #TOK10049 (
  ( #TOK10047 #TOK10046 #TOK10045 )
  ( #TOK10048 ) ) )
#TOK10044 ) 解釈結果の部分木 (TOKEN の木)
( #TOK10043 )
**** RESULT OF INTERPRETATION ( SYMBOL ) ****
( TE I-RU (
  ( WA ROBOT AO-I )
  ( ARU-KU ) ) )
( NI HUKA-TSU TE ) 下線部は、一語として登録されている
( IE )

```

```

**** CATH OF TOK ****
#TOK10049 .... #CATH200005 .... TE I-RU
#TOK10047 .... #CATH200000 .... GA
#TOK10046 .... #CATH100025 .... ROBOT
#TOK10045 .... #CATH100003 .... AO-I
#TOK10048 .... #CATH200003 .... ARU-I
#TOK10044 .... NIL .... NI HUKA-TSU TE
#TOK10043 .... #CATH100028 .... IE
概念アトムが NIL
即ち未知語である

```

(c) 解釈結果

```

**** FRAMES FROM PARTED TREE ****
#FRH1005 .....
((FRAME - CHANGE-LOC)
 (SSLLOT (S1 (#TOK10046 #VTOK1059))
 (S2)
 (S3 (#TOK10048 LOC)) 「ロボットが歩いている」
 (S4 (#TOK10048 COEF)) に対応するフレーム
 (S5 (#TOK10048 DEGREE))
 (S6 (#TOK10048 -))
 (S7 (#TOK10048 XAXIS))
 (S8)
 (S9 (#TOK10048 CONSTANTLY))
 (S10 (#TOK10048 CHANGE))
 (S11)
 (S12))
 (FLAG (SUBJ - S1) 文であることを示す
 (MAIN - T)
 (UPPER (S10 S11 S12) NIL #TOK10049)
 (LOWER)
 (ASPECT - BE)
 (CAUSE - T)))
この他、「青いロボット」「家」
に対応するフレームが作られる
**** RESULT OF GATHERING PARAMETER FROM FRAMES ****
(
 (FRML - ( #FRH1005 #FRH1008 #FRH1009 ) )
 (TOKTL - ( ( #TOK10049 (
 ( #TOK10047 #TOK10046 #TOK10045 )
 ( #TOK10048 ) ) )
 ( #TOK10044 )
 ( #TOK10043 ) ) )
 (TYPE CL) 認識のタイプ
 (SUBJ OBJECT) 複数の意味表現が得られた場合
 (COGSELECT VSELECT) 主語
 (SKELETON ( DO (
 ( = #VTOK1059 )
 ( BE <+-> CAUSE <- FACT <-> DO CHANGE ) ) ) ) )
主動詞を指定する
**** RESULT OF INFERENCE OF UNKNOWN WORD MEANING ****
( #TOK10049 (
 ( #TOK10048 #TOK10044 #TOK10043 ) 画面から得た意味表現
 ( #TOK10047 #TOK10046 ) ) ) により修復された木
推定された未知語の概念
**** UNKNOWN WORD MEANING ****
UNKNOWN-WORD
#TOK10044 .... #CATH200007 .... NI HUKA-TSU TE
..... ( <+-> DO TOV <+ )

```

(d) 入力文の意味推定及び未知語の意味推定

図8 処理例

```

***** ORIGIN CATH ( INPUT ) *****
#CATH200007
SURFACE.. NI HUKA-TSU TE
TREE..... ( <+> DO TOV <+ )
***** RESULT OF PARTITIONING *****
#CATH200009
SURFACE.. TE
TREE..... ( <+> )
#CATH200010
SURFACE.. NI HUKA-TSU
TREE..... ( DO TOV <+ )

***** ORIGIN CATH ( SYSTEM ) *****
#CATH200005
SURFACE.. TE I-RU
TREE..... ( DO BE <+> )
***** RESULT OF PARTITIONING *****
#CATH200008
SURFACE.. I-RU
TREE..... ( DO BE )
#CATH200009
SURFACE.. TE
TREE..... ( <+> )

*** UPDATED SENTENCE *** 単語分解の結果更新された入力文の内容
( #TOK10050 (
  ( #TOK10051 #TOK10048 #TOK10052 #TOK10053 #TOK10043 )
  ( #TOK10047 #TOK10046 #TOK10045 ) ) )
( I-RU (
  ( TE ARU-I TE NI HUKA-TSU IE )
  ( GA ROBOT AO-I ) ) ) 下線部は一語として登録されている
( #TOK10043 #TOK10053 #TOK10052 #TOK10045 #TOK10046
  #TOK10047 #TOK10048 #TOK10051 #TOK10050 )

#TOK10043 --- #CATH100028 --- IE
#TOK10053 --- #CATH200010 --- NI HUKA-TSU
#TOK10052 --- #CATH200009 --- TE
#TOK10045 --- #CATH100032 --- AO-I
#TOK10046 --- #CATH100025 --- ROBOT
#TOK10047 --- #CATH200000 --- GA
#TOK10048 --- #CATH200003 --- ARU-I
#TOK10051 --- #CATH200009 --- TE
#TOK10050 --- #CATH200008 --- I-RU
  
```

単語辞書中に存在する単語「ている」と入力文中に存在する単語「に向かっ」との間で単語分解が起こる。その結果、新たに「に向かっ」と「て」が単語辞書に登録され、入力文では「て」と「いる」に分かれる。

文型を学習し、更新された文型を用いて入力文を(AO-I ROBOT GA IE NI HUKA-TSU TE ARU-I TE I-RU) 発話してみる。

(e) 単語分解及び発話

```

***** STATE OF SYSTEM (SCE MEMORY) ***** 文型学習を行った後の文型記憶の内容 (一部分)
:
#SCE100007
EL.....(#CAT100009 #SCE100011)
      ( ARU-I , TE )
WO.....(#SCE100011 #CAT100009)
#SCE100010
EL.....(#CAT100011 ((#SCE100004) (#SCE100011)))
      ( I-RU ((GA) (TE)))
WO.....(#SCE100004 #SCE100011 #CAT100011)
#SCE100011
EL.....(#CAT100012 ((#SCE100013) (#SCE100007)))
      ( TE ((NI HUKA-TSU) (ARU-I)))
WO.....(#SCE100013 #SCE100007 #CAT100012)
#SCE100013
EL.....(#CAT100013 #SCE100000)
      ( NI HUKA-TSU , IE )
WO.....(#SCE100000 #CAT100013)
:
:
  
```

(f) 学習後の文型の内容

図8 処理例

sequence 1	sequence 2
(1) りんご	(1) <u>赤いりんご</u>
(2) りんご	(2) <u>青いりんご</u>
(3) ロボット	(3) りんごは赤い
(4) りんごがある	(4) りんごは青い
(5) りんごが机の上にある	(5) 3つのりんご
(6) りんごがところがある	(6) 赤いりんごは青いりんごより大きい
(7) ロボットが歩いている	(7) 青いりんごは赤いりんごより小さい
(8) りんごが机の下にある	(8) 赤いりんごの大きさは3つのりんごの中で最大だ
(9) りんごが机の下をところがある	(9) 赤いりんごの大きさは最大だ
(10) りんごが木の前をところがある	(10) 青いりんごの大きさは赤いりんご以下だ
	(11) 赤いりんごは4つのりんごの中で1番大きい
	(12) 青いりんごは4つのりんごの中で2番目に大きい
	(13) 黄色いりんごは4つのりんごの中で3番目に大きい

但し、下線部分は本来複数単語に分かれるべき単語が一語の未知語として入力される例である。

図9 入力文シーケンスの例

6. まとめ

以上、視覚情報を利用して簡単な日本語の意味と文法を学習するシステムの動作について述べた。言語学習システムでは、未知語を含んだ入力文に対応する意味構造を推定する部分が重要であり、本システムでは同時に与える視覚情報から意味表現を推定している。本システムでは、図形について言及する文のみしか与えることができないが、ある程度複雑な文構造を持った文章も学習により理解できるようになる。

謝辞

本稿の作成にあたり協力して頂いた小原研究室の諸氏に感謝します。

参考文献

- [1] Anderson, J.R. : "A Theory of Language Acquisition Based on General Learning Principles", IJCAI-81, pp.97-103, (1981).
- [2] Selfridge, M. : "A Computer Model of Child Language Acquisition", IJCAI-81, pp.92-96, (1981).
- [3] Clark, H.H. and Clark, E.V. : "Psychology and Language", Harcourt Brace Jovanovich (1977).
- [4] 高木, 秋川, 小原 : "簡単な言語を学習する A.I. システム(1)", 信学技報, AL79-92 (1980).
- [5] 高木, 伊東, 北岡, 清水, 小原 : "視覚情報に対する自然語文のつきあわせ処理について", 情処学会 知識工学と人工知能研究 30-5 (1983).
- [6] 高木, 小原 : "属性形容詞の意味構造—意味表現方法の一つの試み—", 信学論 (D), J65-D, 11, pp.1427-1434, (1982).
- [7] 高木, 伊東, 六沢, 北岡, 清水, 小原 : "二次元世界における視覚情報からの日本語文の生成", 信学論 (D), J67-D, 2, pp.216-223, (1984).