

計算機マニュアルを対象とした日英機械翻訳システム における日本語文の解析

長尾真 辻井潤一 森正裕 久米稚子
(京大・工)

1. システム構成

本翻訳システムはトランスファー方式を用い、翻訳の対象を日本語計算機マニュアル(富士通データセットユーティリティ使用手引書)として、英語への翻訳を目的としている。現在、このマニュアル中に含まれる(表や図中の文を除く)すべての文を入力し、翻訳実験を行っている。本報告ではこの中の特に日本語文解析について述べる。

図1に日本語解析過程の流れを示す。

第1ステップはPL/Iによってプログラムされており、入力文より個々の単語を分割し、その品詞・活用形を決定する。ただしこのステップにおいては、「なければならぬ」、「ことができる」などの慣用的な表現は分割せず、1つの単語としている。

第2ステップはLISPによってプログラムされており、第1ステップの単語列から日本語中間表現を生成する。まず図1の(4)、(5)、(6)の部分において入力日本語文中の各述部間の係り受けを決定する。この操作によって入力日本語文は、各文セグメント^(*)の係り受けを表現する木構造に変換される。この文体木は文の大域的な構造を表現すると同時に、以下の文解析の手順を与える。次に(7)において文セグメント内の述部及び各格要素の間の係り受けを表層格照合によって決定し、日本語中間表現を生成する。また(8)において(7)までで生成された中間表現に対して、計算機マニュアル固有の性質を考え、不要な解釈を消去することによって解析の数を減らす。以下、2.において図1の(4)(5)、3.において(7)及び(6)(8)について述べる。

また(8)において(7)までで生成された中間表現に対して、計算機マニュアル固有の性質を考え、不要な解釈を消去することによって解析の数を減らす。以下、2.において図1の(4)(5)、3.において(7)及び(6)(8)について述べる。

(*)文セグメントとは「格要素列+述部」を示す。ただし格要素列はすべて述部の格として係るとは限らない。例えば、「利用者が修正するデータセットを指定する。」では、「利用者が修正する」と「データセットを指定する」の二つの文セグメントに分割できる。

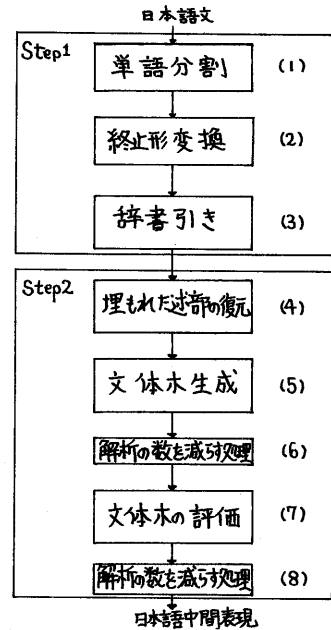


図1 日本語解析過程

2. 文体木

2-1. 文体木の役割と生成

文体木生成ルールは、入力単語列より、述部及び文セグメント間の係りを決める単語(「ため」、「際」、「とき」等)を認識する照合パターンと、それが照合したときに生成される文体木の構造(複数)の対によって記述される。従って、処理する文中にどれだけの述部を許すか、また述部間の係り受け関係にどれだけの種類を許すかによって必要な文体木ルールの数が決まる。

計算機マニュアル文においては、それほど述部の数が多く現われることはなく、現システムでは述部の数が4個までのルール(約50ルール)が用意されている。(このルールによって実験文の約95%に対して文体木が生成できる。)

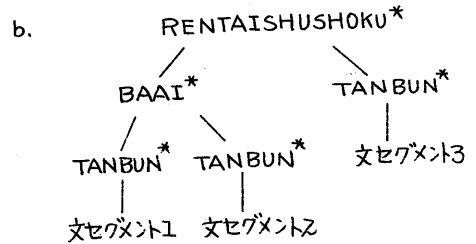
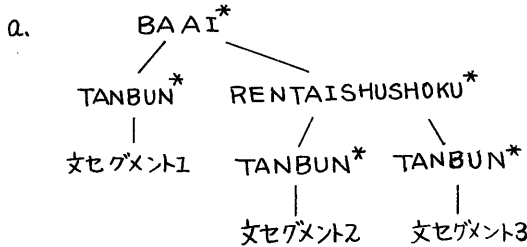
この文体木ルールをうまく使うと分野固有の表現の型を積極的に使って不要な解釈結果が発生するのを防ぐことができる。

例えば、次の表現の場合、

(例1) 利用者がプログラムを実行する場合、次に示すパラメタを指定しな
文セグメント1 文セグメント2

ければならない。
文セグメント3

文体木としては、



(*はプログラム名)

の二つが生成されるが、計算機マニュアルでは、aのみでほとんどの文が正しく解釈される。前述のように文体木は以後の解析の流れを決定するプログラム木としての役割を果たしている。例えば上のaの文体木では、プログラムBAAIは、TANBUN, RENTAISHUSHOKUを起動し、その実行結果に対して「場合」についての処理を行なう。RENTAISHUSHOKUは、二つのTANBUNを起動し、その実行結果について埋め込み操作を実行する。TANBUNは、1つの文セグメントを入力として格要素の認識、述部との係りなどを決定する操作を行なう。

2-2. 埋もれた述部の復元

例えば、

(例2) パラメタがNEWの場合、次の操作が行われる。

(例3) 利用者がデータを修正後、プログラムを実行する。

のような表現を考える。これは、

(例2) パラメタがNEWである場合、次の操作が行なわれる。

(例3) 利用者がデータを修正した後、プログラムを実行する。

の省略形であるが、マニュアル文には非常に多く現われる。このような例では、このまま処理を進めると、それぞれ「パラメタが」、利用者がは文セグメント内の係り先がなくなるので、例2、例3のように埋もれた述部を復元する必要が生じる。

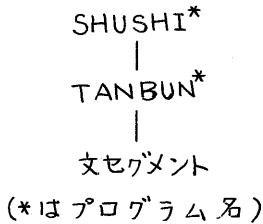
このような前処理を行なう必要のある表現の例を図2に示す。この復元の処理は、ある特定の単語が現われたときにその前後の敬語を調べることにより判断できるので、復元処理の必要な品詞列をパターン集合によって記述しておくことにより処理される。

3. 文の解析

3-1 文セグメント内の処理

文セグメントは、述部1つを持つ日本語文である。入力文中の各文セグメントは、それが入力文中でどのような形で使われていたかに応じて、BAAI, RENYOC HUSHI等の異なるプログラムによって処理される。どの文セグメントをどのプログラムで処理すべきかは、前述の文体木によって示される。最も簡単な例として、単文の場合の入力文を処理する解析の流れを図3に示す。

まず、文体ルールで処理した結果は次のようになる。



次にこのプログラム木が評価される。プログラムSHUSHIは、終止形の述部を持つ文に対して働くもので、一般にすべての文体木のtop nodeとして必ず付加される。図中の(2), (3), (4)の部分はプログラムTANBUNによって処理される。

最初に(5)の部分について説明する。表層格照合は、文セグメント内の格要素(主に'名詞句'+格助詞)が現在処理中の述部と共起し得るかどうかを判定するものであり、述部の辞書に記述された用例パターンと文セグメント内の格要素並びとの間の照合によって行なわれる。この照合の際には、語順は無視される。動詞の用例パターンの記述例を図4に示す。図に示したように、この照合操作は格要素のとり得る助詞の種類、名詞句の主名詞が持つ意味マーカのチェックより成り立っている。この意味マーカは、名詞句処理においても使用されており、現在マニュアル文中の名詞を分類するのに約70種の意味マーカが設定されている。

図3の(2), (3), (4)のステップは、上に述べた照合操作のための準備を行なう。一般に、述部に続く「～てある」「～れる、られる」等の表現は、述部のアスペクトモダル等も変化させると同時に、その述部の表層上の用例パターンを変化させる。図3のステップ(2)は、動詞辞書に記述された複数の用例パターンから、その動詞に続く表現をみて、適切なものを選択するための段階である。現在、動詞辞書には次の4種の用例パターンが記述されている。

入力部分単語列	変更後の部分単語列
(NP) の場合	(NP) である 場合
(サ変名) 後	(サ変名) する た 後
(サ変名) の 後	(サ変名) する た 後
(サ変名) の とき	(サ変名) する とき
(NP) の (サ変名) の後	(NP) の (サ変名) を する た 後
(NP) の (サ変名) の場合	(NP) の (サ変名) を する 場合

(NP) : サ変名詞を含まない名詞句(単語列)
 (サ変名) : サ変名詞
 | : 単語の切れ目を示す。
 活用語は終止形で示してある。

図2 埋もれた述部の復元

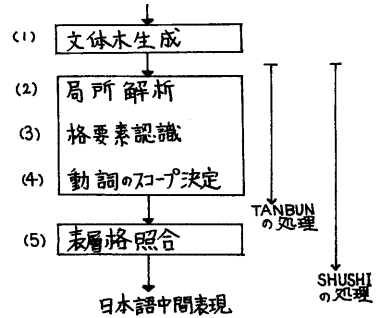


図3 単文の処理

- '実行'+ 'する' — パターン1
- '実行'+ 'する'+ 'れる' — パターン2
- '実行'+ 'する'+ 'できる' — パターン3
- '実行'+ 'する'+ 'てある' — パターン4

この用例パターンの選択を行なうための記述(例えば, 'できる'の右側がサ変名詞ならばサ変名詞をサ変動詞に変え, パターン3を選ぶ)は, それぞれの単語('できる', 'てある', 'れる' 等)の辞書記述中に示されている。

動詞等続く表現は, 用例パターンを変化させるものの他に, 単にアスペクト, 様相, 態だけを変化させるものもあり, この種のものもこの処理ステップで同時に処理される。

図3の(3)のステップは'名詞'+ '格助詞'という単語列が現われたときにそれを1つの格要素として認識するためのステップである。また, (4)は照合操作のスコープを機械的に決定するステップであり,

例えば,

格要素1 + 格要素2 + 述部

のときには,

(格要素1 + 格要素2 + 述部)

格要素1 + (格要素2 + 述部)

格要素1 + 格要素2 + (述部)

〔ただし,
()は述部のスコープを示す。〕

の3通りのスコープで次の照合操作をとるべきことが示される。この3種のスコープのうち2種はこの文セグメント内では係りが決定されない格要素が存在する解釈に対応し, プログラム SHUSHI では格要素についての助詞がほ以外の場合は失敗と判断する。文体ホヤに示される文セグメント処理用プログラムには, この SHUSHI 以外に RENTAISHUSHOKU, BAAI, RENTYCHUSHI 等がある。RENTAISHUSHOKUの動作については, 文献りて報告したので, 次の各節では RENTYCHUSHI, BAAI 等のプログラムについて述べる。

3-2 文セグメント間の関係

文の接続関係を従来の文法により整理すると次のようになる。

1. 埋め込み構造
2. 連用中止法
3. 接続助詞による接続 ('て', 'が', 'は' 等)
4. 特別な名詞による接続 ('場合', '際' 等)

我々のシステムにおいては, この4種類の接続関係を接続される文間の結合の強さに注目し, 類似な解析方法を必要とする次の4種類に分類している。以下この分類の2, 3について, それぞれの解析過程を述べる。

- パターン1 → MP150*
- パターン2 → MP152*
- パターン3 → MP154*
- パターン4 → MP150*, MP154*

	MP150*		
((*P1	CAMAC11	SM141	SM142)
((*P2	CAMAC12	SM143	SM144)
),			
	MP152*		
((*P1	CAMAC13	SM141	SM142)
((*P2	CAMAC11	SM143	SM144)
),			
	MP154*		
((*P1	CAMAC13	SM141	SM142)
((*P2	CAMAC11	SM143	SM144)
),			

CAMAC11 : (が)

CAMAC12 : (を)

CAMAC13 : (によって, て)

SM141~144 は意味マカの集合がセットされる。

マッチすると, (JIKKOSURU *P1 *P2) の *P1, *P2 の位置に格要素がセットされる。

図4 用例パターンの例 ('実行する')

1. 埋め込み構造

2. 連用中止法及び接続助詞‘で’

3. ‘で’以外の接続助詞及び特別な名詞による接続

3-3 連用中止法の処理及び接続助詞‘で’の処理

連用中止法及び接続助詞‘で’による文の接続は、接続される2文間の結合が強く、多くの表層上の構造が共有される。したがってこの共有関係を解析時に復元する必要がある。問題点を整理すると以下ようになる。

問題点1 省略された格要素の補充

(例4) 利用者がデータを指定し、(利用者が) プログラムを実行する。

問題点2 省略された助動詞の補充

例えば次のような文を考える。

(例5) 利用者がパラメタを指定し、プログラムを実行しなければならぬ。

(例6) 利用者がパラメタを指定でき、プログラムを実行できなければならぬ。

(例7) 利用者がパラメタを指定して、プログラムを実行しなければならぬ。

(例8) データが修正して、入力してある。

(例9) 利用者がパラメタを指定して、プログラムを実行しない。

例5は

(例5’) 利用者がパラメタを指定しなければならぬ。

(例5’’) 利用者がプログラムを実行しなければならぬ。

の2文を連用中止法で表現したと考えられ、‘なければならぬ’は‘指定する’の後で省略されている。例6も同様である。また例7では、‘なければならぬ’が‘指定する’の後で省略されているとも考えられるし、そうでないとも考えられる。例8では‘てある’が修正する’の後で省略されているが、この場合はこの省略の補充がないと、前半が意味的に解釈できなくなる。(すなわち‘てある’の補充によって修正する’の用例パターンが変更される。)例9は、‘ない’について省略があるとも考えられるし、そうでないとも考えられる。

これらの現象は、末尾の助動詞等の種類、並列文の接続の種類によって異なり、図5に示すような関係がある。

問題点3 連用中止法又は‘で’の意味的なつながり方の解析

(例10) 利用者がパラメタを指定してプログラムを実行する。

(例11) この表現を用いて、パラメタを指定する。

例10では並列にな、ている動作が時間的には、同時・順序的の2通りの解釈がある。

また例11は前半の部分が後半の内容に対して手段の意味を持っている。

	連用中止法	‘で’
受身	X	X
‘できる’	△	X
使役	X	X
‘なければならぬ’	△	△

△: 左側の文セグメントで省略されている可能性のあることを示す。

X: 左側の文セグメントで省略されることがないことを示す。

図5 助動詞等の省略

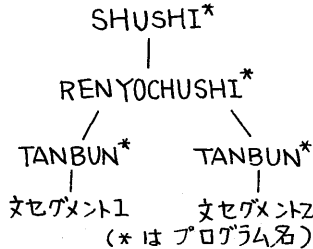
上記のうち、問題点3は各文セグメントの意味的内容を深く認識する必要があり、

またマニュアル中の文では'and'で英文を生成してもほとんど問題がないので、現システムでは扱わない。以下簡単な例に基づいてプログラムの処理を説明する。

次の例に示す文が入力されると、

(例12) 利用者がプログラムを指定し , 実行できる。
文セグメント1 文セグメント2

文体系として次のものが生成される。



この文体系が評価されると、SHUSHIはRENYOCHUSHIを起動する。このプログラムRENYOCHUSHIの処理の流れを図6に示す。以下この図を説明する。

Step(1): 述部の単語リストを作成する。例12の場合には、

(指定 'する' * '実行' 'できる')
 のようなリストが作られる。ここで*は文セグメントの区切りを示す記号である。この操作は照合パターンによって実行される。

Step(2): 問題点2のための処理を行なう。この処理を行なうための照合パターンにより、単語の消去、補充を決定する。例えば照合パターンに、

「サ変名詞+ 'する' + * + 4変名詞+ 'できる' ならば、
 「サ変名詞+ 'できる' + * + サ変名詞+ 'できる' に変更する。

という記述があれば、上記の述部リストは、

(指定 'できる' * '実行' 'できる')

に変更される。

Step(3): Step(2)により、変更された述部リストをもとにして入力文セグメントの各単語列の変更を行なう。従って上の例では、

(例12') 利用者がプログラムを指定でき、実行できる。

のように変更される。

Step(4) (省略された格要素の復元): 左側の文セグメント内の格要素をすべて右側の文セグメントに加えて表層格照合を行い、マッチすれば左側の格要素すべてが右側の格要素として補充されたことになる。もしマッチしなければ、右側文セグメントに移した格要素を減らして順次同様の操作を行う。

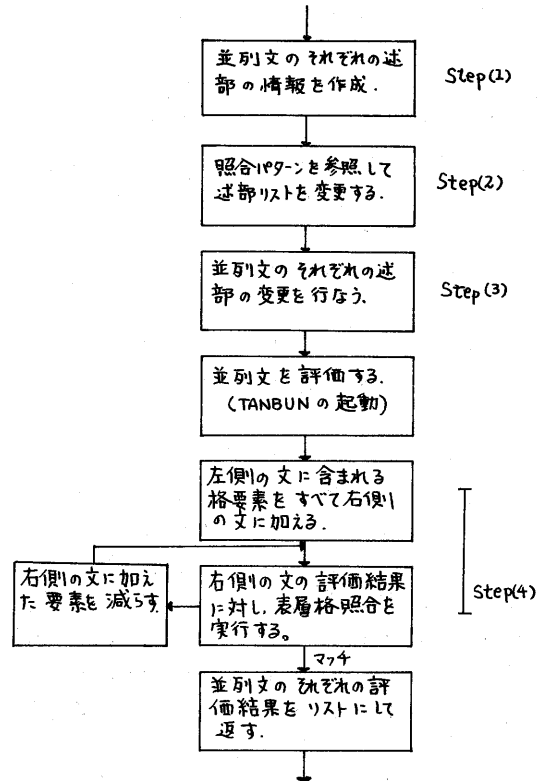


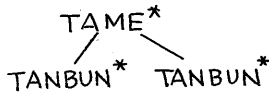
図6 連用中止法・'て'の処理

3-4 特別な名詞による接続及び'て'以外の接続助詞による接続の処理

この接続形式は条件、逆接、時間的前後関係等表現する接続形式で前のタイプの接続に比べると、接続される文の独立度が高く、共有構造はあまりない。例えば連用中止法で見られる助動詞の省略はなく、格要素の省略においても、もしあっても意味的に省略されていることが多い。従って連用中止法の場合のような表層格照合の処理ではこの省略を補うことはできない。

この接続形式を解析するための処理の流れは図7のようになる。

中間表現の top node の決定とは入力文セグメントがどのような意味的關係で接続されているかを決定する操作をいう。一般に連用中止法の場合とは違って、このタイプの接続は意味的關係が接続語によって明示されているので、比較的容易に文関係を認識できる。しかしながら文体木が、



のようなときには、2つの文セグメントの間には、目的・根拠の2つの意味的關係を示している可能性がある(それぞれ英語においては異なる表現形をとらなければならない)。本システムにおいては左側の文セグメントの述部のアスペクトを参照することによって比較的簡単に意味的關係が決定できる。この決定は主に左側の文セグメントの述部が動作を示しているか、状態を示しているかによって決定され、'ため'に関しては動作のとき目的、状態のとき根拠とする。(図8参照)

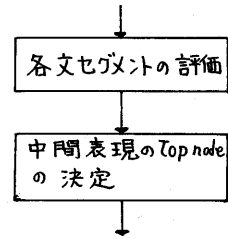


図7 特別な名詞による接続'て'以外の接続助詞による接続の処理

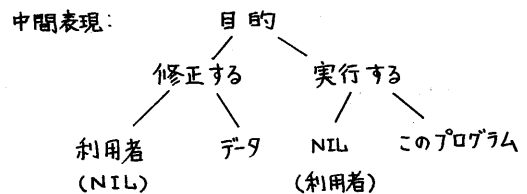
3-5 解析の数を減らすための処理

上までに述べた文体木生成・解析の段階では、それぞれ動詞・格要素の係り受け等によって、複数の解釈がある可能性がある。これらの複数の解釈を減らすための処理として、

- ・連用中止法の連用形の述部はできるだけ同じ様態(動作・状態或)を持った述部に係る。
- ・副助詞'しか'を伴った格要素は否定を示す助動詞・形容詞'ない'を持っている述部に係る。
- ・連用中止、'場合'の接続関係が一文内に現れるときは、文体木において、RENYOCHUSHIはBAAIの下部nodeとなる。

などが考えられるが、本システムでは文体木生成後と解析終了後において、

入力文: 利用者がデータを修正するため、このプログラムを実行する。



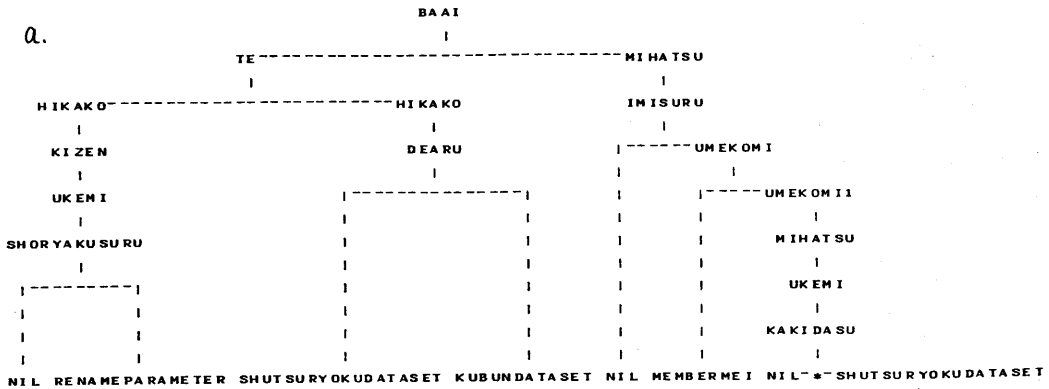
入力文: 利用者がデータを修正する場合、このプログラムを実行する。



図8 接続関係の表現

各現象ごとのプログラムが起動できるステップを設定している。

4. 実験結果

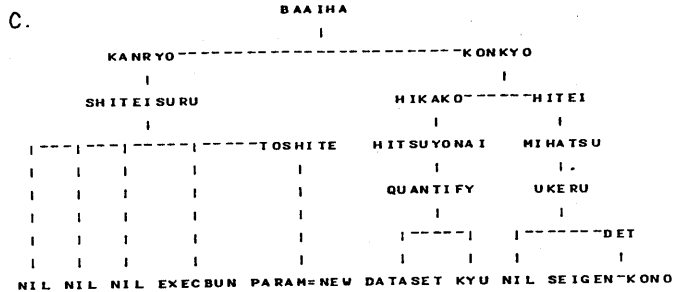
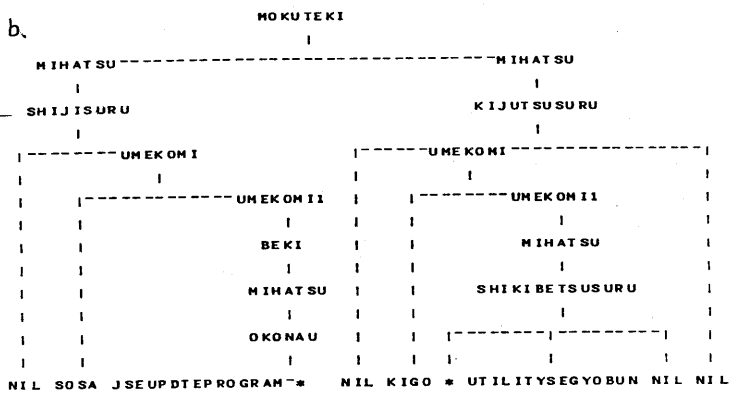


それぞれの入力日本語文を次に示す。

a. RENAMEパラメタが省略されていて出カデータセットが区分データセットの場合，出カデータセットに書き出されるメンバ名を意味する。

b. JSEUPDTEプログラムが行うべき操作を指示するため，ユーティリティ制御文を識別する記号を記述する。

c. EXEC文で PARAM=NEWと指定した場合は，旧データセットは必要ないため，この制限は受けない。



現システムは次のような規模をもっている。

名詞：約400項目，動詞・形容詞・形容動詞：約150項目

モーダル・アスペクト等：約150項目 文体ルール 約50ルール

(参考文献)

- 1) 長尾, 辻井, 三田村, 平川, 又米: 日英機械翻訳システムの言語現象からみた問題, 情報処理学会第21回全国大会, 1980
- 2) M. Nagao, J. Tsujii, K. Mitamura, H. Hirakawa, M. Kume: A Machine Translation System From Japanese Into English, COLING 80, pp. 414-423, 1980
- 3) 長尾, 辻井, 畑崎: 計算機マニュアルを対象とした日英機械翻訳システムにおける英文の生成, 第25回計算言語学研究会, 1981