機械翻訳の一模型*

西村恕彦**

まえがき

筆者の属する研究室では機械翻訳に関する研究を続けているが，最近になって一般化された文法向き翻訳の実験プログラムを試作し，英文和訳，和文英訳の初期実験を行なった[1]．

この実験（および現在加えつつある改訂を含めて）に用いられた模型およびプログラムについて，いわば超文法（meta grammar）とでもいうべき立場から報告する．すなわち英語とか日本語とか特定の自然言語の文法を解説することは目的ではない．それは例としてはひかれるが，ねらいはいろいろの自然言語の文法を記述するのに必要で共通な最小限の規約つまり超文法の一例を報告することにある．

この超文法によって，広い範囲の自然言語の文法が容易に記述できるものと期待している．もちろんその記述はただちに計算機にかけて検証できる．この実験プログラムの主な用途は狭義の構文分析にあり，自然言語の構文の大部分を句構造文法により，一部分を変形規則によって記述し，その記述の妥当性を機械的に検証することにある．

われわれの研究室では日本語および英語における文法的事象すなわち，語彙，語形変化，品詞，構文などに関する調査と研究を進めている．またより実際的な実用的な翻訳システムについても研究開発を進めている．それらについてはここではまったく触れない．

1. システムの構成

この翻訳システムはほぼ次のような構成を有している．初期実験ではこれらすべてに穿孔カードが用いられたが，将来は辞書と資料とは漢字テレタイプの紙テープになるであろう．

プログラム
|
文法表
|
辞書
|
資料

これらを計算機に与えると資料の原文が翻訳され，訳文が漢字テレタイプの紙テープとして得られる．

最初に稼動したプログラムは NEAC 2200 COBOL でコーディングされ，約 190 個の手続き命令を含んでいた．

1.1 辞書

辞書は次のような構成になっている．すなわち，見出し語と訳語と品詞とが一組になって一項目を作る．この項目が不定数個集まって辞書となる．項目数はいわゆる語彙に相当する．これは数千以下でも実験的にはかなり良い翻訳ができる．しかし実用的には少なくとも数万は必要で，そうなると普通の計算機の記憶装置には収まらなくなり，大容量記憶装置に収容し，うまく呼出す方策を研究せねばならない．この初期実験ではたかだか数百語を収容する記憶容量を用意できただけで，これでは辞書とよぶことはできず，いちいちの資料にあわせた単語帳であるにすぎない．

	見出し語 72ビット	訳語 72ビット	品詞 12ビット
数百～数万項目	a		冠
	and	そして	接
	any	如何なる	形
	by	による	前
	card	カード	名
	cat	猫	名
	cut	切る	他

各項目の大きさにはとりあえず一定の制限を設けてある．見出し語は 72 ビット以内，訳語も 72 ビット以内である．これはホレリス符号を用いた場合には 12 字に相当し，英語についてはかなり楽に多くの単語を含められる．ローマ字化した日本語ではそれほど楽ではない．漢テレ符号を用いた場合には 6 字に相当し，日本語では楽だが，英語には苦しい．

品詞は 12 ビットで，6 ビット 2 字に符号化するかあるいは外部媒体上ではまったく記号化しておいて読込

---

* A Model of a Machine Translation, by Hirohiko Nisimura (Electrotechnical Laboratory)

** 通産省電気試験所情報処理特別研究室

み時に変換するかする．品詞は語の品詞と句の品詞とに区別がないので，数が増えることを予想して 12 ビットを用意し，約4,000種まで可能とした．実験によれば品詞の数はそこまでは増えそうもない．たとえば8ビット程度でも何とかなるかもしれない．しかし品詞を表現するビット長を変えるとシステム全般にわたる手直しが必要になるから，そのような事態がおこらぬよう十分な余裕をみておくほうがよかろう．

品詞を漢テレ符号（12ビット/1字）で直接表現することも可能である．ただしそれは品詞の数が増えた場合には著しく煩わしくなる．

なお特に英語では多品詞語や多義語がきわめて多い．実用的な翻訳のためにはこの手当をしておかねばならないが，本システムではこの点は目をつぶっている．すなわち，ひとつの見出し語はただひとつの品詞と訳語だけを有する．

1.2 訳 語

翻訳の終った文は辞書から取出された訳語と，若干の挿入語との列として出力される．出力文の語と語とのあいだには何も区切り記号は置かれない．たとえば英文和訳のときは訳文はきっちりつまった漢字仮名交り文として印刷され，分かち書きはされない．和文英訳のときは英語の語と語とのあいだには原則として間隔が必要である．これは表のように辞書に登録する訳語の語頭に空白をつけておくことで大部分の問題が解決される．

見出し語	訳 語
家	␣house
木	␣tree
青い	␣blue
た	ed
。	.

現在のプログラムは活用や語尾変化に関する特別の手続きを含んでいないので，単純な連結（catenation）だけでは表現できないような問題ではうまくゆかないことがある．たとえば

愛し＋た → love＋ed → loveed → loved

のような手順が必要なのだが，その最終段階は組込んでないからさしあたってはシステムの利用者が，辞書の段階などでなんらかの手当をするしかない．

1.3 品 詞

プログラム（つまり超文法）の立場からは品詞は次のものが区別される．

（a） 普通の品詞
（b） 終止符の品詞
（c） 空品詞
} 辞書で与える

（d） 挿入語の品詞
（e） 句の品詞
} 文法表で与える

（a） 普通の品詞はたとえば英文法でいう八品詞のようなものにほぼ相当する．資料の原文中の語の文字列をひとつの品詞でおきかえる．普通の品詞は次々に品詞の列につながれ，品詞列としての文を構成する．

（b） 終止符の品詞はこの翻訳システムが文単位の翻訳方式をとっているために必要なもので，語彙分析の段階から構文分析の段階に切替える働きだけをもっている．すなわち文字列の照合は左から右へ進み，得られた品詞が終止符の品詞として登録されているものと一致すれば，語彙分析はいったん中止される．終止符の品詞はただ一種とはかぎらない（疑問符，感嘆符など）．

（c） 空品詞を指定される語は，文字列として辞書の見出し語に登録され資料と照合はとれるが，構文分析や合成の段階において品詞や訳語を与える必要のないものである．具体的に英文和訳の例では，単語間の普通の空白や，三人称単数現在動詞および複数名詞の語尾のsがあげられる．これらは資料の文字列を構成する要素で，かつ語として識別される．しかし，その品詞は品詞列につながれることはない．

（d） 挿入語の品詞は syntactic role indicator に相当するもので，たとえば

名詞＋他動詞＋名詞
→ 名詞＋**主格**＋他動詞＋**目的格**＋名詞

というように，ある品詞の列で決定される構文中の句の文法的な機能（syntactic role）を明示するためのものである．これは合成の段階でも用いられることがあり，そのときは訳語をも必要とする．たとえば

主格 “が”
目的格 “を”．

（e） 句の品詞はある品詞列のなかに句構造をなす部分が見出されたときにそれをおきかえるものである．これはいわゆる句構造分析の代表的な操作である．

これらの品詞のうち，(c) 空品詞以外の4者はすべて品詞列につながれる．構文分析の段階で文法表の見出しと照合をとるときには，それらのあいだには何の区別もなされない．たとえば語の品詞と句の品詞とのあいだに区別はない．

1.4 文 法 表

文法表は次のような構成になっている．すなわち，見出しと処理指定とが一組になって一項目を作る．この項目が不定数個集まって文法表となる．項目数はこのシステムによる構文規則の数に相当する．最初の実験では 30 個弱の規則を用いた．

見出し 72ビット			処理指定 72ビット			
品詞列		合成	品詞	訳語	位置	長さ	操作
名名	→		名		0	2	置換
名	→		句		0	1	置換
句前句	→	逆順	句		0	3	置換
句述	→		主	が	1		挿入
他句	→		目	を	1		挿入

見出しは品詞の列である．大きさはさしあたり 72 ビット以内に制限してある．つまり1個の品詞は 12 ビットで表現されるから6個の品詞からなる列まで識別できる．システムによってはこれを2個までに制限している例がある．しかしそれでは文法を記述するのに不自然になることがある．たとえば，

名詞句＋前置詞＋名詞句 → 名詞句

のような構文規則の左辺は3個の品詞で記述するのがもっとも素直であろう．英文和訳の場合は上限が4個程度で間に合うのではないかと思われた．

処理指定は 72 ビットの固定長の情報で，その内部は次のように割振ってある．

合成の語順	18ビット	（1ビット）
句の品詞	12ビット	（5ビット）
挿入訳語	24ビット	（2ビット）
処理位置	6ビット	（2ビット）
処理長さ	6ビット	（3ビット）
操作	6ビット	（2ビット）

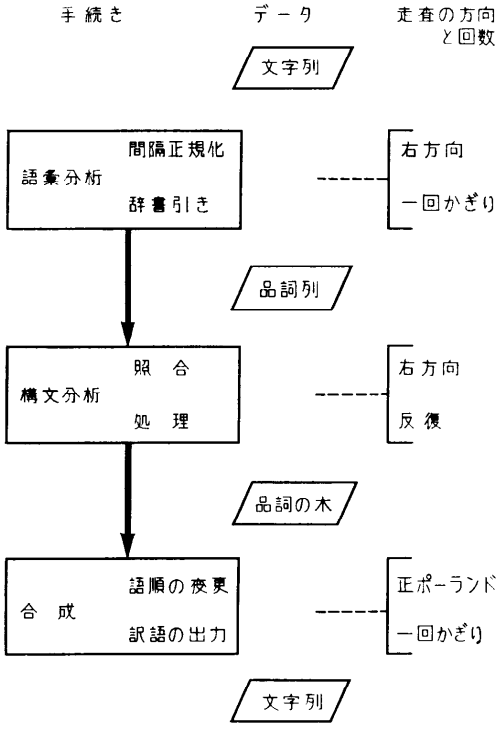
明細の説明は別に行なうが，この割振りは非常に冗長である．当初は相当に複雑な，また広範囲の処理指定を予想していた．実際には初期の実験ではるかに単純な処理しか必要でなかった．そのとき用いた情報量を上表の右端に括弧で示した．

2. 翻訳手続き

翻訳は次の図に示すような手順で進められる．その具体的な算法については，大部分はすでに報告済みである[2]．しかし近い将来，このシステムの開発がいちおう終った時期に，より的確な算法を全体として報告する機会を持ちたいと願っている．

2.1 文と語

資料すなわち翻訳すべき原文は 12 ビットの文字の列として穿孔符号化される．この文字列は間隔の正規化を行なって読込まれる．正規化を行なうのは，英語のように分かち書きのされている原文中で，間隔をまたいだ熟語があるのを識別するためである．



原資料　… case␣␣by␣case …
正規化　… case␣by␣case …
辞書　　case␣by␣case

だから間隔をまたいだ熟語の存在を認めないか，あるいは原資料の穿孔において間隔が正規化されている保障があればこの手続きは省略できる．

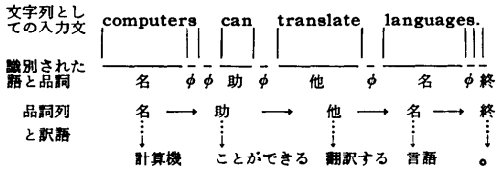
正規化された資料は語に分割される．語という言葉は間隔で仕切られること，語尾変化，共通の語幹などは何も意味しない．それは単に文字の列であって，辞書に登録されている見出し語の文字列と照合のとれるものを指すにすぎない．この定義は先述した，間隔をまたいだ熟語や，日本語のように分かち書きのされていない資料を処理するのには有利である．

語への分割は (a) 左から右へ，(b) 辞書の見出し語の文字列と照合のとれる，(c) 最長の文字列を識別同定して行なわれる．資料の処理走査は終止符に行き当るまで続く．終止符は語のひとつであって，辞書の見出し語と照合のとれた文字列について，その品詞(終止符の品詞)によって識別される．終止符の次の語から終止符までの資料を文と呼ぶ．略式な定義を与えて説明の助けとする．

資 料 ::＝文 | 資料 文
文 ::＝終止符 | 語 文
終止符 ::＝語
語 ::＝文字 | 語 文字

この簡単なモデルによって，日本文の分かち書きの問題と，語尾変化の問題との双方を解決できるものと期待している．たとえば変化語尾は単に語として識別され，品詞を与えられて構文分析の段階に渡される．

再説すると，資料の文は文字の列として受取られ，同定できる文字列つまり語に分割される．語はそれぞれ品詞におきかえられ，次に品詞の列につながれる．この，文字列を入力とし，品詞列を出力とする手続きは語彙分析の段階とよんでもよい．



2.2 品詞列の照合

構文分析の前半は照合で，後半は照合のとれた部分にたいする所要の処理である．照合は原文の品詞列と文法表の見出しとのあいだで右方向，最長一致にもとづいて行なわれる．この手続きは語彙分析における辞書引きとほとんど同じである．

文法表の各項目のあいだには明白な優先順位は何もない．システムによっては，文法表を分割して優先権をあたえたり，各項目に優先番号をつけたり，あるいは有限状態オートマトンの推移図のように照合がとれたときに次に引くべき表を指定したりする方式がある．われわれのシステムにはそういう仕掛けが一切ないので，場合によっては自然言語の構文を記述するのにちょっとした技巧を要することがある．そのかわり文法表の項目を書く順位はどうでもよく，また改訂増補が著しく容易になる．

自然言語の各種の構文規則のあいだに優先順位や適用条件の制限があるときには，次のいずれかの方策によって解決できよう．

（a） 前後の文脈（すなわち前後の品詞列）を十分長くとって見出しの照合のとれる条件を限定する．

（b） 長い見出しが結果として優先的に照合される．

（c） ダミーの品詞を挿入するか，またはおきかえる．

しかしこの点については経験を積んでいないのでこれ以上の議論は避けたい．

2.3 品詞列の処理—超文法

文法表の各項目は見出しと処理指定とが対になっている．ある品詞列が見出しに同定されると，指定された処理がそこに加えられる．処理の種類としてどういうものを採用するかは，この翻訳システムの性格，能力を大きく左右する．ここでは4種類の基本的な処理をとった．これらが本システムにおける狭義の超文法になる．

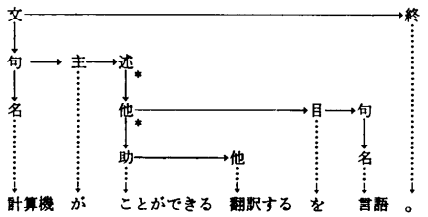
（a） 句構造のおきかえはいわゆる句構造分析に対応する．ただし，見出しと照合のとれた品詞列のなかの指定された品詞列をいわば context sensitive なかたちでおきかえることができる．おきかえられた句構造は合成の段階で展開するときに語順を変更できる．

（b） 変形指定はいわゆる transformation rule に対応する．見出しと照合のとれた品詞列のなかの指定された部分の語順を転倒する．

（c） 挿入は見出しと照合のとれた品詞列のなかの指定された位置に，ひとつの品詞（およびそれにぶらさがった葉の文字列）を割り込ませる．これはある句構造がその前後の文脈中で有する文法的な機能（syntactic role）を明示するのに用いる．この機能，たとえば格（case）は英語では語順であらわされ，日本語では文字列である助詞であらわされる．

（d） 飛越しは句構造の結合の強さの優先順位を処理する目的で導入された．品詞列と見出しとの照合の反復は原則として文頭に戻って繰り返えされる．ところがこの飛越し操作が指定されると，次の照合は品詞列の指定された部分からはじまる．

自然言語の構文は上記の4種の超文法のみによって記述せねばならない．文法表の項目の大部分は，そしてたいていの文は（a）の句構造のおきかえによって構成されている．さきの例文の構文分析がすべて終った形は次のとおりである．



2.4 合 成

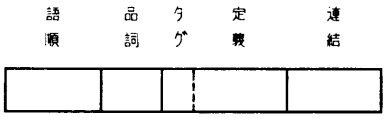
構文分析は品詞列のなかに文法表の見出しと照合のとれる品詞列がひとつも見出だせなくなったところで

終る．このとき全体が文としてまとまっていなくても局所的には構文分析ができている可能性がある．この初期実験では構文分析が終ったところで，何もチェックをしないでそのまま合成出力の段階に移る．

より進んだシステムでは多品詞語などについてさまざまな品詞の組み合わせを試みて，最終的に文としてまとまるものだけを出力する例がある．

さて合成出力の手続きは，品詞の木構造を正ポーランド記法の順に2回走査して行なわれる．この時期には構文分析の段階のような反復走査はなされない．したがって繰り返えして，ある処理を加えたいような場合には，その処理要素を構文分析の段階に含めねばならない．

品詞の木構造はいわゆる相続順位リストであらわされている．このリストの要素つまり節（node）の構造は次のとおりである．



72 ビット

合成の語順	18 ビット
品詞	12 ビット
定義つなぎのタグ	6 ビット
定義つなぎ	18 ビット
連結つなぎ	18 ビット

第1回の走査は語順の変更で，すべての節を規則的に走査し，そのうち語順の転倒の指定されている節について，それに直接従属する句構造の語順を変更する．語順指定はいまのところ正順と逆順の二通りしか区別していないが，構文分析の変形（transformation）の手続きを利用すれば，いま少し多様な語順変更も可能であろう．

第2回の走査は訳語の出力で，すべての節を規則的に走査して，そのうち葉に相当するもの，すなわち直接に文字列を指している節だけを左から右に順次ひろいだし，その指している文字列を出力する．

ある節のなかの定義つなぎのタグは，その節の定義つなぎの内容が次のいずれであるかを識別する．

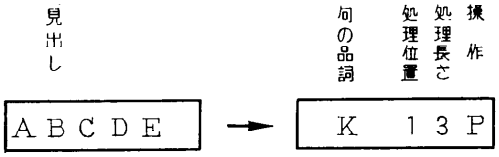
- 他の品詞の節を指す
- 文字列つまり葉を指す
  - 辞書の項目番号
  - 作業記憶中の場所
  - リテラル

3. 超文法の利用法

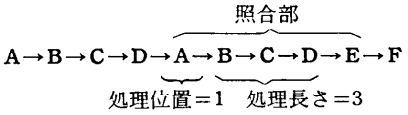
本システムはわずか四つの超文法規則を有するにすぎない．それらを用いて自然言語の構文規則をどう記述するかを例をひきながら説明する．

3.1 句構造のおきかえ

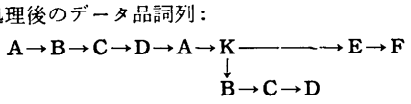
例 1. AとEとにはさまれたBCDという品詞列があったら，それをKでおきかえる．



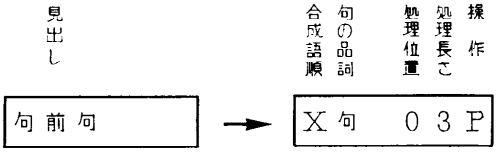
処理前のデータ品詞列：



処理後のデータ品詞列：



例 2. 名詞句＋前置詞＋名詞句を名詞句でおきかえる．ただし合成出力のさいは語順を転倒させる．

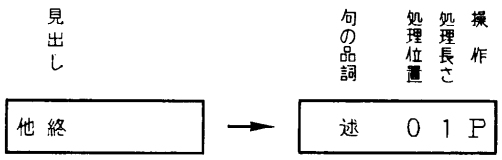


処理前のデータ品詞列：

処理後のデータ品詞列：

この例ではおきかえた名詞句の節には，のちに利用されるはずの合成語順に関する指定の情報が保持されている．図ではそれを星（＊）印で示した．

例 3. 他動詞のあとに（目的格たるべき名詞なしに）いきなり終止符がきたら，他動詞を述語でおきかえる．

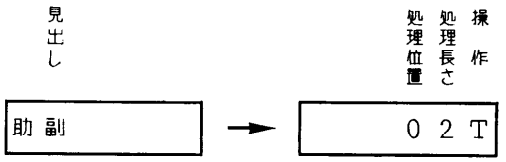


処理前のデータ品詞列：　照合部
句→他→終
処理後のデータ品詞列：　句→述→終
他

これらの句構造のおきかえにおいては，合成語順はさしあたり正順が逆順かの二通りだけである．処理位置は0または正の整数，処理長さは1以上の正の整数である．

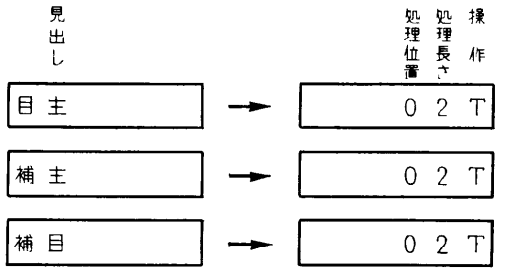
3.2 変形指定

例 4. 助動詞の後の副詞を前にだす．



処理前のデータ品詞列：　句→助→副→他→名→終
処理後のデータ品詞列：　句　助←副　他→名→終

例 5. 和文英訳において，原文の主格名詞句，目的格名詞句，補格名詞句の語順が任意のときに，この順にそろえなおす．

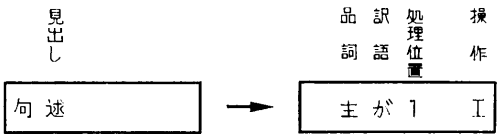


処理前のデータ品詞列：　目→補→主→動→終
目　補　主　動→終
処理後のデータ品詞列：　目→補　主　動→終

これら変形指定では，処理位置は0または正の整数，処理長さは2またはそれより大きい整数である．

3.3 挿　入

例 6. 名詞句と述語とのあいだに主格をあらわすロール・インジケータを挿入する．このロール・インジケータは出力文では文字列「が」であらわす．

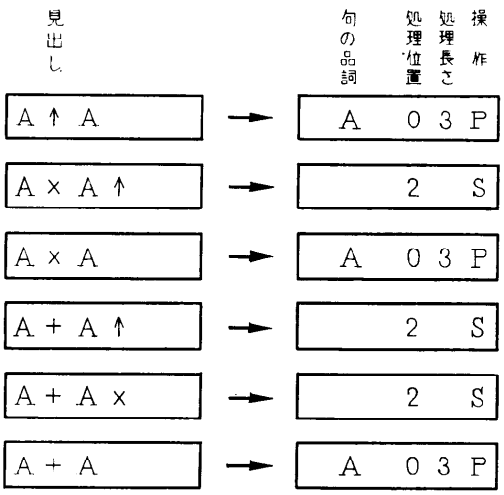


処理前のデータ品詞列：　句→述→終
処理後のデータ品詞列：　句→主→述→終
が

処理位置は0または正の整数である．0は照合のとれた品詞列の直前の位置を指す．

3.4 飛　越　し

例 7. 普通の数式における加算（+），乗算（×），冪（↑）の各符号の結合の強さを評価する．



飛越し（S）の処理位置は1以上の正の整数である．

4. 翻　訳　例

初期実験に用いた入力資料はホレリス・カードに穿孔された英文および日本文（ローマ字）である．その翻訳された出力は漢字テレタイプの紙テープで取出され，オフ・ラインで漢字仮名交り文または大文字のみの英文に印字された．その例を示す．ただしこの例の一部は予定された出力であって，かならずしも実証されたものではない．

a translation experiment using
kanji-type of electrotechnical-lab.
電気試験所の漢テレをもちいた翻訳実験.
computer can translate the language.
計算機が言語を翻訳することができる.
the computer can translate.
計算機が翻訳することができる.
when the cat met the tiger
he wanted to run away.
猫が虎を見たとき彼が逃げ去ることを望んだ.
the cat he wanted to eat
run fast.
彼が食べることを望んだ猫が速く走る.
use of magnetic tape for data
storage in the algol.
ALGOLにおけるデータ記憶のための磁気テープの利用.
otogasindoudearukotowa
dokusyawayokusitteiru.
␣READER␣WELL␣KNOW␣THAT␣SOUND
␣IS␣VIBRATION.
sonotenkeitekinaatudenhandoutaiwa
ryuukakadomiumudearu.
␣THE␣TYPICAL␣PIEZO␣SEMICONDUCTOR
␣IS␣CdS.

参考文献

1) 西村恕彦：漢テレをもちいた翻訳の予備実験，情報処理学会機械翻訳研究委員会，1966-5.
2) 西村恕彦：木表現とリスト処理の算法，第7回プログラミング—シンポジウム報告集，1966-1.

（昭和41年10月8日受付）