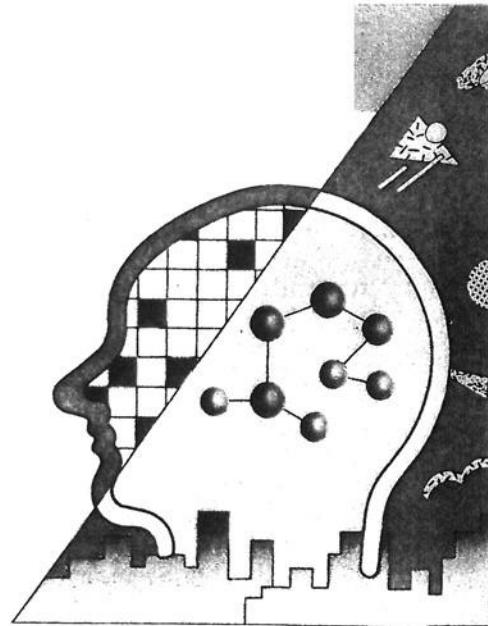


Prologによる機械翻訳システム

その2

高野 真

今回は、Prologによる機械翻訳システムの第2回ということで、英文の解析から日本語の生成までのアルゴリズムを詳しく考察する。さらに、日本語の文生成について具体的なインプリメントに取りかかることにする。



英日翻訳のアルゴリズムは、大きく分けて下記に示す3段階がある。

- ①形態素解析と辞書引き
- ②構文解析と中間表現の生成
- ③中間表現から日本語を生成

①と②についての本格的な解説は次回以降に行なう。ここでは簡単にふれるだけにとどめる。③については今回具体的なシステムを参照しながら、詳しく考察することにする。

英日翻訳システムを作成する立場から①、②、③を重要性の順番に並べれば、①=②>③の順番になるであろう。翻訳は①からシーケンシャルに②③へと進んで行くので、解析の部分で失敗すると、日本語の生成の部分にいくら優れたものをもってきても翻訳全体が失敗してしまうからである。そこで、今回の連載でも①および②の解析の部分はあと回しにして、一番簡単な③の部分から考えていくことにする。

AI 形態素解析と辞書引き

形態素解析では、スペースで区切られた各単語の品詞や意味を辞書で引くことが第1目的となる。後々の処理とのつながりから、熟語の処理や"I'm"、"can't"といった省略表現の展開なども、ここで行なわれることがある。

各単語を引くといっても英単語の綴りは状況によって様々に変化する。それに応じていちいち辞書を記述するのではメモリ効率がかなり低下してしまう。そこで、我々が辞書を引くときと同様に単語を原型に直してから引くという作業が必要になってくる。

この、原型に直すという作業は以外にやっかいなものである。主なものを以下に示す。

- ①名詞の複数形や所有格
- ②動詞の3人称単数現在の「s」
- ③動詞の過去形あるいは過去分詞
- ④動詞の現在分詞
- ⑤形容詞の比較級や最上級
- ⑥副詞の比較級や最上級
- ⑦全ての品詞に共通な項目として文頭の大文字の処理

どの処理をとっても「単語の最後が短母音+子音のときは最後の子音を重ねる」といった様々な例外規則があり一筋縄ではいかない。そのほか細かいものとしては、"Mr."のような省略表現"1987"のような数字などの処理などが挙げられる。

形態素解析は次に来る辞書引きと密接に関係している。たとえば "one hundred and eighty-three" のように一定の規則にしたがって生成される数詞の場合、形態素解析は "eighty-three" を "83" と解析するための解析プログラムを持っている必要がある。また "hundreds of" のように原型に直すと意味が分からなくなってしまう熟語にも対応できるようにしなければならない。

辞書引きは形態素解析ができていれば、単に辞書を引くだけであるが、以下の 2 つの問題を解決しなければならない。

①熟語への対応

②同じ綴りで品詞や意味の異なる単語に対応する

熟語処理は、構文解析の手法を用いて複数の単語の並びに対応できるようにすればよい。同綴意語の処理は、後に続く構文解析で失敗したときに後戻りしてきた時に別の辞書項目を返すようになればよいだろう。

AI 構文解析と中間表現の生成

構文解析は、機械翻訳の質を決定する最も重要な部分である。今回の翻訳方式は、トランスファー方式を採用する。中間言語としては格文法(英日共通)を基にした表現に落とす事を目標にする。ただし、今回は純粋のトランスファー方式ではなく多少中間言語方式に近ものに変形している。

先月号で紹介したオーソドックスなトランスファー方式を図示すると図 1 のようになる。2 重線で囲ってあるのは今回のシステムと本来のトランスファー方式とで大きく異なっている部分である。

英文の解析の流れは、トランスファー方式そのものである。すなわち、辞書引き、構文解析、(意味解析)が行なわれる。しかし、①で示しているように解析で落とす中間表現は日本語の格文法である。したがって、英語の中間表現から日本語の中間表現に変換する手続きは必要なくなる。

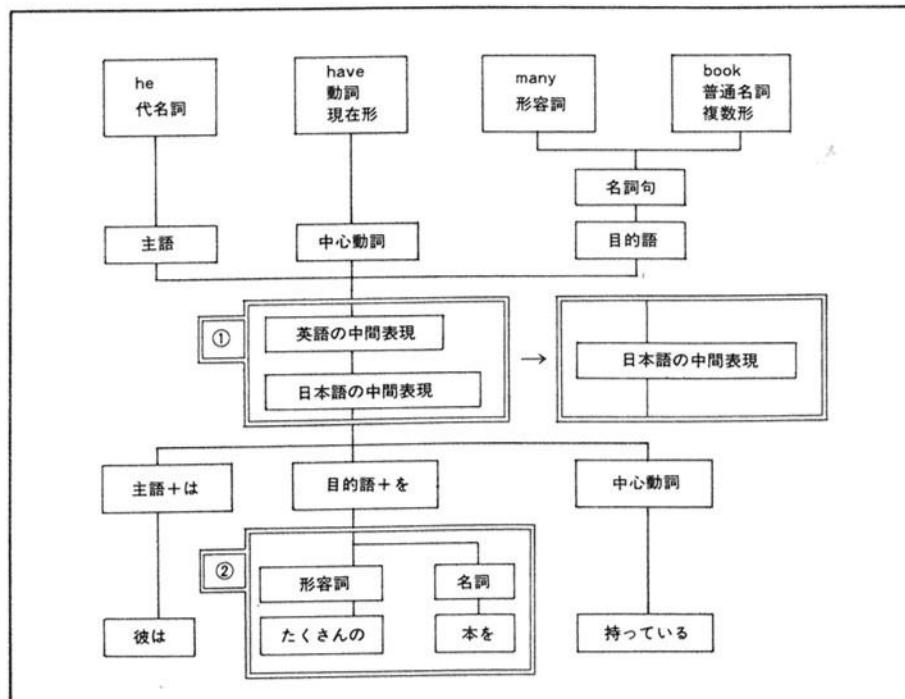


図 1 トランスファー方式による翻訳の流れ

英文の解析の後はすぐ、日本語の中間表現から日本語を合成する段階に移る。

②で示しているように「形容詞」という品詞は日本語の中間表現から省かれている。そのかわり「たくさんの本」で一つの文法カテゴリーを形成するようにとらえるようになっている。この用にすることで、翻訳文の質をたいして落とさずに、規則を減らすことができる。今回は、中学生用の薄い参考書から規則を抽出するようにしているが、それでも規則の数は 100 のオーダーとなる。

次に、意味処理の問題を考えよう。以下のような文があるとする。

My brother is playing baseball.

私の兄が野球をしています（私の兄は野球をする事です）

My hobby is playing baseball.

私の趣味は野球をする事です（私の趣味が野球をしている）

カッコ内のような誤訳を防ぐためには、「野球をするのは人間である」という情報が必要になる。こうした情報を翻訳システムに組み込むには、

①名詞辞書に「Brother- 男」といった情報を付加する。

②動詞がどの様な格をとるかを辞書に記述しておく。たとえば "play" という動詞の辞書には、主語は「生物」であり、目的語は「スポーツやゲームや楽器」であるといった情報が付加されている。

③解析の際に、動詞の格情報と名詞の意味カテゴリーとのマッチングを取る。そのときには「男は人間であり、人間は生物である」といった意味カテゴリー同士の関係も考慮する。

といった、手続きが必要となる。今回のシステムでは、こうした意味の問題もできるだけ取り込んでいきたい。

AI 中間表現から日本語を生成

日本語の文生成は今回と次回に分けてインプリメントと解説を行なうことにする。今回は動詞を中心とした述語の合成、次回は埋め込み文の処理を考慮した日本語全体の生成および機械翻訳システムとの結合について考える。

日本語の文生成の手続きの中で手間のかかるのは、動詞を中心とした述語の生成である。述語の生成さえできれば、名詞句や副詞、埋め込み文の処理はそう難しくはない。特に日本語の用言の場合は、次に続く語によって様々な活用語尾を取るために、そのためのルールは膨大なものになる。

今回紹介する述語合成ルーチンは、本来の物からかなり簡略化してあるがそれでも相当量のルールを含んでいる。なお、今回インプリメントするシステムは「日本語情報処理」（電子通信学会）の第 8 章を参考にしているので詳しい説明はそちらの方を参照していただきたい。

まず、述語の合成の例として、以下の文を考えてみよう。

述語の合成では、上に挙げた例に見られるように

動詞 + 受身使役 + アスペクト & テンス + 判断 + 判断のテンス + 働きかけ
というシーケンスが考えられる。

動詞とは「動詞の語幹 + 語尾」のことで語尾の活用形は後に続く語に支配されている。

「彼は飛行機で飛んでいたらしいので、....」
飛ん でいた らしい ので
動詞 アスペクト&テンス 判断 働きかけ

「トムは、ベンに殴られたようだった。」
殴 られ た ようだ った
動詞 受身 アスペクト&テンス 判断 判断のテンス

図 2

受身使役は

「れる・られる」「せる・させる」

という助動詞で表わされる。「せる」「れる」は5段活用やさ変活用の未然形に、「させる」「られる」は上一段、下一段、か変の未然形に接続する。

アスペクトは、英語でいう過去完了形で表わされる表現で、完了や継続、結果などを表わすための語である。アスペクトは次のテンスと密接に結びついている。テンスは、過去・現在・未来という時制を表わしている。アスペクト&テンスの部分には

「た・だ」「ている・でいる」「ていた・でいた」「つつある」「つつあった」

という語がくる。アスペクトとテンスに応じてどの様な付属語を取るかは、動詞固有のアスペクト素性によって決められる。物事が進行中である事を表わすのに

酔いがさめつつある

という表現は取れるが

酔いがさめている（完了を表わしている）

という表現は誤りである。

「判断」の部分には「ない（否定）」や「らしい（推定）」「たい（希望）」といった機能を表わすいわゆる助動詞がくる。「判断のテンス」ということで助動詞の部分の時制が規定されることもある。

最後の「働きかけ」の部分には「命令」「疑問」「禁止」といった終助詞を初めとする助詞で表現される語がくる。助詞や助動詞といった付属語は、前に来る語の品詞や活用形を規定するので、述語の合成は語尾の方から行なわなければならない。活用形の規定は、～行の～段活用の～形というだけでなく音便形の選択などの問題も考慮しなければならない。音便形の選択を考慮しないと、

彼は取りた魚を近所に配りた

といったおかしな日本語文が生成されてしまう。

以上述べた点を考慮した述語生成システムがリスト1に挙げるDCG文法のリストおよびリスト2のプログラムである。

AI 実行例

述語を生成させるには

述語生成(_辞書,_命題,_アスペクト,T1,Mode,T2,Func).

を実行すればよい。_辞書の部分は要素を4つもった動詞辞書の内容が入る。第1要素は

[[・・・|T],T],

であり、動詞の語幹部分を重リストで表わした物がくる。・・・の部分は動詞語幹をシフトJISコードに変換した物が入る。たとえば「話す」の語幹は「話」であるので

[-265261X],X],

を語幹として登録する第2要素は、

[行,段]

で動詞の活用形が入る。第3要素は、現在は無名変数 "_" でよい。将来的には動詞の必要とする格に関する情報がここにはいる。第4要素はアスペクト素性を表わす。アスペクト素性は現在は

[+,+][+,-]

[-,+][-,-]

の4通りであり、それぞれ以下に挙げるような特性を持っている。

- [+,+] 「～ている」がつかない動詞
- [+,-] 「前,間,後,ながら」がつかない動詞
- [-,-] 「～し終える」がつかない動詞
- [-,+] その他大半の動詞はここにはいる

これらを実行すればよい。_ 命題の部分は

- [] 受身
- 可能 使役

のうちのどれかである。以下共通だがデフォルトは[] (ヌルリスト) である。_ アスペクトの部分には

- []
- 未完了 (アスペクト素性が [+,+], [+,-], [-,+] のとき)
- 完了
- 未完了反復 (アスペクト素性が [-,-] のとき)
- 未完了結果 (アスペクト素性が [-,-] のとき)

のうちのいずれかがはいる。T1 はテンスを表わし、

- []
- 過去

の2つのいずれかがはいる。[] (ヌルリスト) は非過去 (現在あるいは未来) を表わしている。アスペクト素性が [+,+] の場合の未完了過去、完了過去、未完了非過去、完了非過去、[+,-] の場合の完了過去、中立=[] には対応する表現は無い事に注意していただきたい。

Mode には

- []
- 打ち消し・・・・(ない)
- 希望・・・・・・(たい)
- 推定・・・・・・(らしい)
- 様態・・・・・・(そうだ)
- 伝聞・・・・・・(そうだ)
- 不確定・・・・・・(ようだ)
- 丁寧・・・・・・(ます)
- 意志・・・・・・(う・よう)
- 打ち消し意志・・(まい)

のいずれかがはいる。T2 は上記の助動詞の時制であるが動詞のテンスと同様

- []

過去

のうちのどちらかが入る。最後の Func には

- []
- 連言 順接
- 選言 逆接
- 例示 並列
- 程度 命令
- 帰着点 疑問
- 類推 禁止
- 限定 確認
- 仮定 平叙
- 原因

のいずれかが入る。

例として、

話す

述語生成([

[-26526|X],X],

[さ,5],

-'

[-,+]

],[],完了,過去,推定,[],[]).

来る

述語生成([

[X,X],

[か,変],

-'

[-,-]

],[使役,[],[],[],[],命令).

編む

述語生成([

[-27182|X],X],

[ま,5],

-'

[-,+]

],[],[],過去,[],[],[]).

今回挙げた述語生成ルーチンは、「活用形が後に続く語に規定される」という用言の特性のために、語尾の方から生成するようになっている。逆に文節の頭の動詞の方からマッチングを取るようすれば、日本語向けの述語解析ルーチンとしてリスト1はそのまま用いる事もできる。日英翻訳システム等を作成するときには多少の拡張を行なえば、述語から必要な情報を抽出する事に使えるであろう。

次回は、今回のルーチンを用いて格文法表現及び意味ネットワークからの日本語生成システムを紹介する。意味ネットワークからの文生成は今回の機械翻訳システムとは直接の関係は無いが、コンサルテーションシステムの出力文の合成など様々な分野に応用がきくために、特別につけ加えることにする。

%*****% % か 5 % % 咲く % %*****%	%*****% % ま 5 % % 飲む % %*****%
か 5 ([-320871_1], _1, 未然). か 5 ([-320791_1], _1, 未然 5). か 5 ([-320851_1], _1, 連用). か 5 ([-320631_1], _1, 連用 音便). か 5 ([-320941_1], _1, 連用 音便). か 5 ([-320831_1], _1, 終止). か 5 ([-320831_1], _1, 連体). か 5 ([-320941_1], _1, 連体). か 5 ([-320811_1], _1, 仮定). か 5 ([-320811_1], _1, 命令).	ま 5 ([-320361_1], _1, 未然). ま 5 ([-320321_1], _1, 未然 5). ま 5 ([-320351_1], _1, 連用). ま 5 ([-320151_1], _1, 連用 音便). ま 5 ([-320341_1], _1, 終止). ま 5 ([-320341_1], _1, 連体). ま 5 ([-320331_1], _1, 仮定). ま 5 ([-320331_1], _1, 命令).
%*****% % が 5 % % 騒ぐ % %*****%	%*****% % ら 5 % % 乗る % %*****%
が 5 ([-320861_1], _1, 未然). が 5 ([-320781_1], _1, 未然 5). が 5 ([-320841_1], _1, 連用). が 5 ([-320941_1], _1, 連用 音便). が 5 ([-320821_1], _1, 終止). が 5 ([-320821_1], _1, 連体). が 5 ([-320801_1], _1, 仮定). が 5 ([-320801_1], _1, 命令).	ら 5 ([-320251_1], _1, 未然). ら 5 ([-320211_1], _1, 未然 5). ら 5 ([-320241_1], _1, 連用). ら 5 ([-320631_1], _1, 連用 音便). ら 5 ([-320231_1], _1, 終止). ら 5 ([-320231_1], _1, 連体). ら 5 ([-320221_1], _1, 仮定). ら 5 ([-320221_1], _1, 命令).
%*****% % 貸す % %*****%	%*****% % わあ 5 % % 思う % %*****%
さ 5 ([-320771_1], _1, 未然). さ 5 ([-320691_1], _1, 未然 5). さ 5 ([-320751_1], _1, 連用). さ 5 ([-320731_1], _1, 終止). さ 5 ([-320731_1], _1, 連体). さ 5 ([-320711_1], _1, 仮定). さ 5 ([-320711_1], _1, 命令).	わあ 5 ([-320191_1], _1, 未然). わあ 5 ([-320881_1], _1, 未然 5). わあ 5 ([-320941_1], _1, 連用). わあ 5 ([-320631_1], _1, 連用 音便). わあ 5 ([-320921_1], _1, 終止). わあ 5 ([-320921_1], _1, 連体). わあ 5 ([-320901_1], _1, 仮定). わあ 5 ([-320901_1], _1, 命令).
%*****% % 立つ % %*****%	%*****% % あ下一 % % 考える % %*****%
た 5 ([-320671_1], _1, 未然). た 5 ([-320581_1], _1, 未然 5). た 5 ([-320651_1], _1, 連用). た 5 ([-320631_1], _1, 連用 音便). た 5 ([-320621_1], _1, 終止). た 5 ([-320621_1], _1, 連体). た 5 ([-320601_1], _1, 仮定). た 5 ([-320601_1], _1, 命令).	あ下一 ([-320901_1], _1, 未然). あ下一 ([-320901_1], _1, 連用). あ下一 ([-320901_1], _1, 終止). あ下一 ([-320901_1], _1, 連体). あ下一 ([-320901_1], _1, 仮定). あ下一 ([-320901_1], _1, 命令).
%*****% % 死ぬ % %*****%	%*****% % か下一 % % 助ける % %*****%
な 5 ([-320561_1], _1, 未然). な 5 ([-320521_1], _1, 未然 5). な 5 ([-320551_1], _1, 連用). な 5 ([-320151_1], _1, 連用 音便). な 5 ([-320541_1], _1, 終止). な 5 ([-320541_1], _1, 連体). な 5 ([-320531_1], _1, 仮定). な 5 ([-320531_1], _1, 命令).	か下一 ([-320811_1], _1, 未然). か下一 ([-320811_1], _1, 連用). か下一 ([-320811_1], _1, 終止). か下一 ([-320811_1], _1, 連体). か下一 ([-320811_1], _1, 仮定). か下一 ([-320811_1], _1, 命令).
%*****% % 飛ぶ % %*****%	%*****% % が下一 % % 投げる % %*****%
ば 5 ([-320501_1], _1, 未然). ば 5 ([-320381_1], _1, 未然 5). ば 5 ([-320471_1], _1, 連用). ば 5 ([-320151_1], _1, 連用 音便). ば 5 ([-320441_1], _1, 終止). ば 5 ([-320441_1], _1, 連体). ば 5 ([-320411_1], _1, 仮定). ば 5 ([-320411_1], _1, 命令).	が下一 ([-320801_1], _1, 未然). が下一 ([-320801_1], _1, 連用). が下一 ([-320801_1], _1, 終止). が下一 ([-320801_1], _1, 連体). が下一 ([-320801_1], _1, 仮定). が下一 ([-320801_1], _1, 命令).

リスト 1 述語生成ルーチンの辞書

%*****% % さ下一 % % 任せる % %*****%	%*****% % ま下一 % % 攻める % %*****%
さ下一([-32071,_1],_1,未然). さ下一([-32071,_1],_1,連用). さ下一([-32071,-320231,_1],_1,終止). さ下一([-32071,-320231,_1],_1,連体). さ下一([-32071,-320221,_1],_1,仮定). さ下一([-32071,-320211,_1],_1,命令). さ下一([-32071,-320261,_1],_1,命令). %*****% % ざ下一 % % 混せる % %*****%	ま下一([-320331,_1],_1,未然). ま下一([-320331,_1],_1,連用). ま下一([-32033,-320231,_1],_1,終止). ま下一([-32033,-320231,_1],_1,連体). ま下一([-32033,-320221,_1],_1,仮定). ま下一([-32033,-320211,_1],_1,命令). ま下一([-32033,-320261,_1],_1,命令). %*****% % ら下一 % % 流れる % %*****%
ざ下一([-32070,_1],_1,未然). ざ下一([-32070,_1],_1,連用). ざ下一([-32070,-320231,_1],_1,終止). ざ下一([-32070,-320231,_1],_1,連体). ざ下一([-32070,-320221,_1],_1,仮定). ざ下一([-32070,-320211,_1],_1,命令). ざ下一([-32070,-320261,_1],_1,命令). %*****% % た下一 % % 立てる % %*****%	ら下一([-320221,_1],_1,未然). ら下一([-320221,_1],_1,連用). ら下一([-32022,-320231,_1],_1,終止). ら下一([-32022,-320231,_1],_1,連体). ら下一([-32022,-320221,_1],_1,仮定). ら下一([-32022,-320211,_1],_1,命令). ら下一([-32022,-320261,_1],_1,命令). %*****% % あ上一 % % 怪いる % %*****%
た下一([-320601,_1],_1,未然). た下一([-320601,_1],_1,連用). た下一([-32060,-320231,_1],_1,終止). た下一([-32060,-320231,_1],_1,連体). た下一([-32060,-320221,_1],_1,仮定). た下一([-32060,-320211,_1],_1,命令). た下一([-32060,-320261,_1],_1,命令). %*****% % だ下一 % % 出る % %*****%	あ上一([-320941,_1],_1,未然). あ上一([-320941,_1],_1,連用). あ上一([-32094,-320231,_1],_1,終止). あ上一([-32094,-320231,_1],_1,連体). あ上一([-32094,-320221,_1],_1,仮定). あ上一([-32094,-320211,_1],_1,命令). あ上一([-32094,-320261,_1],_1,命令). %*****% % か % % 起きる % %*****%
だ下一([-320591,_1],_1,未然). だ下一([-320591,_1],_1,連用). だ下一([-32059,-320231,_1],_1,終止). だ下一([-32059,-320231,_1],_1,連体). だ下一([-32059,-320221,_1],_1,仮定). だ下一([-32059,-320211,_1],_1,命令). だ下一([-32059,-320261,_1],_1,命令). %*****% % な下一 % % 審る % %*****%	か上一([-320851,_1],_1,未然). か上一([-320851,_1],_1,連用). か上一([-32085,-320231,_1],_1,終止). か上一([-32085,-320231,_1],_1,連体). か上一([-32085,-320221,_1],_1,仮定). か上一([-32085,-320211,_1],_1,命令). か上一([-32085,-320261,_1],_1,命令). %*****% % が上一 % % 逃げる % %*****%
な下一([-320531,_1],_1,未然). な下一([-320531,_1],_1,連用). な下一([-32053,-320231,_1],_1,終止). な下一([-32053,-320231,_1],_1,連体). な下一([-32053,-320221,_1],_1,仮定). な下一([-32053,-320211,_1],_1,命令). な下一([-32053,-320261,_1],_1,命令). %*****% % は下一 % % 経る % %*****%	が上一([-320841,_1],_1,未然). が上一([-320841,_1],_1,連用). が上一([-32084,-320231,_1],_1,終止). が上一([-32084,-320231,_1],_1,連体). が上一([-32084,-320221,_1],_1,仮定). が上一([-32084,-320211,_1],_1,命令). が上一([-32084,-320261,_1],_1,命令). %*****% % ざ上一 % % 取じる % %*****%
は下一([-320421,_1],_1,未然). は下一([-320421,_1],_1,連用). は下一([-32042,-320231,_1],_1,終止). は下一([-32042,-320231,_1],_1,連体). は下一([-32042,-320221,_1],_1,仮定). は下一([-32042,-320211,_1],_1,命令). は下一([-32042,-320261,_1],_1,命令). %*****% % ば下一 % % 並べる % %*****%	ざ上一([-320741,_1],_1,未然). ざ上一([-320741,_1],_1,連用). ざ上一([-32074,-320231,_1],_1,終止). ざ上一([-32074,-320231,_1],_1,連体). ざ上一([-32074,-320221,_1],_1,仮定). ざ上一([-32074,-320211,_1],_1,命令). ざ上一([-32074,-320261,_1],_1,命令). %*****% % た上一 % % 落ちる % %*****%
ば下一([-320411,_1],_1,未然). ば下一([-320411,_1],_1,連用). ば下一([-32041,-320231,_1],_1,終止). ば下一([-32041,-320231,_1],_1,連体). ば下一([-32041,-320221,_1],_1,仮定). ば下一([-32041,-320211,_1],_1,命令). ば下一([-32041,-320261,_1],_1,命令). %*****%	た上一([-320651,_1],_1,未然). た上一([-320651,_1],_1,連用). た上一([-32065,-320231,_1],_1,終止). た上一([-32065,-320231,_1],_1,連体). た上一([-32065,-320221,_1],_1,仮定). た上一([-32065,-320211,_1],_1,命令). た上一([-32065,-320261,_1],_1,命令).

(次ページに続く)

```

%*****%
%    な上一 %
%    煮る %
%*****%
な上一([-320551_1],_1,未然).
な上一([-320551_1],_1,連用).
な上一([-32055,-320231_1],_1,終止).
な上一([-32055,-320231_1],_1,連体).
な上一([-32055,-320221_1],_1,仮定).
な上一([-32055,-320211_1],_1,命令).
な上一([-32055,-320261_1],_1,命令).

%*****%
%    は上一 %
%    干る %
%*****%
は上一([-320481_1],_1,未然).
は上一([-320481_1],_1,連用).
は上一([-32048,-320231_1],_1,終止).
は上一([-32048,-320231_1],_1,連体).
は上一([-32048,-320221_1],_1,仮定).
は上一([-32048,-320211_1],_1,命令).
は上一([-32048,-320261_1],_1,命令).

%*****%
%    ば上一 %
%    延びる %
%*****%
ば上一([-320471_1],_1,未然).
ば上一([-320471_1],_1,連用).
ば上一([-32047,-320231_1],_1,終止).
ば上一([-32047,-320231_1],_1,連体).
ば上一([-32047,-320221_1],_1,仮定).
ば上一([-32047,-320211_1],_1,命令).
ば上一([-32047,-320261_1],_1,命令).

%*****%
%    ま上一 %
%    試みる %
%*****%
ま上一([-320351_1],_1,未然).
ま上一([-320351_1],_1,連用).
ま上一([-32035,-320231_1],_1,終止).
ま上一([-32035,-320231_1],_1,連体).
ま上一([-32035,-320221_1],_1,仮定).
ま上一([-32035,-320211_1],_1,命令).
ま上一([-32035,-320261_1],_1,命令).

%*****%
%    ら上一 %
%    燐りる %
%*****%
ら上一([-320241_1],_1,未然).
ら上一([-320241_1],_1,連用).
ら上一([-32024,-320231_1],_1,終止).
ら上一([-32024,-320231_1],_1,連体).
ら上一([-32024,-320221_1],_1,仮定).
ら上一([-32024,-320211_1],_1,命令).
ら上一([-32024,-320261_1],_1,命令).

%*****%
%    さ変 %
%*****%
さ変([-320771_1],_1,未然).
さ変([-320751_1],_1,未然).
さ変([-320751_1],_1,連用).
さ変([-32073,-320231_1],_1,終止).
さ変([-32073,-320231_1],_1,連体).
さ変([-32073,-320221_1],_1,仮定).
さ変([-32071,-320261_1],_1,命令).
さ変([-32075,-320211_1],_1,命令).

%*****%
%    か変 %
%*****%
か変([-320791_1],_1,未然).
か変([-320851_1],_1,連用).
か変([-32083,-320231_1],_1,終止).
か変([-32083,-320231_1],_1,連体).
か変([-32083,-320221_1],_1,仮定).
か変([-32079,-320261_1],_1,命令).
か変([-32079,-320211_1],_1,命令).

%*****%
%    形容詞語尾 %
%*****%
形容詞語尾([-32087,-320211_1],_1,未然).
形容詞語尾([-320831_1],_1,連用).
形容詞語尾([-32087,-320631_1],_1,連用音便).
形容詞語尾([-320941_1],_1,終止).
形容詞語尾([-320941_1],_1,連体).
形容詞語尾([-32081,-320221_1],_1,仮定).

%*****%
%    形容動詞語尾 %
%*****%
形容動詞語尾([-32066,-320211_1],_1,未然).
形容動詞語尾([-32066,-320631_1],_1,連用音便).
形容動詞語尾([-320591_1],_1,連用).
形容動詞語尾([-320561_1],_1,終止).
形容動詞語尾([-320561_1],_1,連体).
形容動詞語尾([-32056,-320251_1],_1,仮定).

%*****%
%    使役 %
%*****%
せる([-32071_1],_1,未然).
せる([-32071_1],_1,連用).
せる([-32071,-320231_1],_1,終止).
せる([-32071,-320231_1],_1,連体).
せる([-32071,-320211_1],_1,命令).
せる([-32071,-320261_1],_1,命令).
させる([-32077,-320711_1],_1,未然).
させる([-32077,-320711_1],_1,連用).
させる([-32077,-32071,-320231_1],_1,終止).
させる([-32077,-32071,-320231_1],_1,連体).
させる([-32077,-32071,-320211_1],_1,命令).
させる([-32077,-32071,-320261_1],_1,命令).

%*****%
%    受身可能 %
%*****%
れる([-320221_1],_1,未然).
れる([-320221_1],_1,連用).
れる([-32022,-320231_1],_1,終止).
れる([-32022,-320231_1],_1,連体).
れる([-32022,-320221_1],_1,仮定).
れる([-32022,-320211_1],_1,命令).
れる([-32022,-320261_1],_1,命令).
られる([-32025,-320221_1],_1,未然).
られる([-32025,-320221_1],_1,連用).
られる([-32025,-32022,-320231_1],_1,終止).
られる([-32025,-32022,-320231_1],_1,連体).
られる([-32025,-32022,-320221_1],_1,仮定).
られる([-32025,-32022,-320211_1],_1,命令).
られる([-32025,-32022,-320261_1],_1,命令).

%*****%
%    打ち消し %
%*****%
ない([-32056,-32087,-320211_1],_1,未然).
ない([-32056,-320831_1],_1,連用).
ない([-32056,-32087,-320631_1],_1,連用音便).
ない([-32056,-320941_1],_1,終止).
ない([-32056,-320941_1],_1,連体).
ない([-32056,-32081,-320221_1],_1,仮定).

%*****%
%    希望 %
%*****%
たい([-32067,-32067,-320211_1],_1,未然).
たい([-32067,-320831_1],_1,連用).
たい([-32067,-32087,-320631_1],_1,連用音便).
たい([-32067,-320941_1],_1,終止).
たい([-32067,-320941_1],_1,連体).
たい([-32067,-32081,-320221_1],_1,仮定).

%*****%
%    推定 %
%*****%
らしい([-32025,-32075,-320831_1],_1,連用).
らしい([-32067,-32075,-32087,-320631_1],_1,連用音便).
らしい([-32025,-32075,-320941_1],_1,終止).
らしい([-32025,-32075,-320941_1],_1,連体).
らしい([-32025,-32075,-32081,-320221_1],_1,仮定).

```

```

%*****%
% 様態 %
%*****%
そうだ([-32069,-32092,-32066,-32021|_1],_1,未然).
そうだ([-32069,-32092,-32059|_1],_1,連用).
そうだ([-32069,-32092,-32055|_1],_1,連用).
そうだ([-32069,-32092,-32066,-32063|_1],_1,連用音便).
そうだ([-32069,-32092,-32066|_1],_1,終止).
そうだ([-32069,-32092,-32056|_1],_1,連体).
そうだ([-32069,-32092,-32056,-32025|_1],_1,仮定).

%*****%
% 不確定 %
%*****%
ようだ([-32026,-32092,-32066,-32021|_1],_1,未然).
ようだ([-32026,-32092,-32059|_1],_1,連用).
ようだ([-32026,-32092,-32055|_1],_1,連用).
ようだ([-32026,-32092,-32066,-32063|_1],_1,連用音便).
ようだ([-32026,-32092,-32066|_1],_1,終止).
ようだ([-32026,-32092,-32056|_1],_1,連体).
ようだ([-32026,-32092,-32056,-32025|_1],_1,仮定).

%*****%
% 丁寧 %
%*****%
ます([-32036,-32071|_1],_1,未然).
ます([-32036,-32075,-32027|_1],_1,未然).
ます([-32036,-32075|_1],_1,連用).
ます([-32036,-32073|_1],_1,終止).
ます([-32036,-32073|_1],_1,連体).
ます([-32036,-32073,-32022|_1],_1,仮定).
ます([-32036,-32075|_1],_1,命令).
ます([-32036,-32071|_1],_1,命令).

%*****%
% 意志 %
%*****%
う([-32092|_1],_1,終止).
よう([-32026,-32092|_1],_1,終止).

%*****%
% 打消し意志 %
%*****%
まい([-32036,-32094|_1],_1,終止).

%*****%
% 命題変換処理 %
%*****%
% [ ]
%*****%
命題変換(_3,_3,[ ],[_1,_2,Kt],[_1,_2,Kt]).
%*****%
% 受身 %
%*****%
命題変換(_2,_1,受身,[さ,変,未然],[φ,下一,Kt]) :-  
れる(_2,_1,Kt).
命題変換(_2,_1,受身,[_3,5,未然],[φ,下一,Kt]) :-  
れる(_2,_1,Kt).
命題変換(_2,_1,受身,[か,変,未然],[φ,下一,Kt]) :-  
られる(_2,_1,Kt).
命題変換(_2,_1,受身,[_3,下一,未然],[φ,下一,Kt]) :-  
られる(_2,_1,Kt).
命題変換(_2,_1,受身,[_3,上一,未然],[φ,下一,Kt]) :-  
られる(_2,_1,Kt).

%*****%
% 可能 %
%*****%
命題変換(_2,_1,可能,[さ,変,未然],[φ,下一,Kt]) :-  
れる(_2,_1,Kt).
命題変換(_2,_1,可能,[_3,5,未然],[φ,下一,Kt]) :-  
れる(_2,_1,Kt).
命題変換(_2,_1,可能,[か,変,未然],[φ,下一,Kt]) :-  
られる(_2,_1,Kt).
命題変換(_2,_1,可能,[_3,下一,未然],[φ,下一,Kt]) :-  
られる(_2,_1,Kt).
命題変換(_2,_1,可能,[_3,上一,未然],[φ,下一,Kt]) :-  
られる(_2,_1,Kt).

```

(次ページに続く)

```

%*****%
% 使役 %
%*****%
命題変換(_2,_1,使役,[さ,変,未然],[φ,下一,Kt]) :-  

    せる(_2,_1,Kt).  

命題変換(_2,_1,使役,[_3,5,未然],[φ,下一,Kt]) :-  

    せる(_2,_1,Kt).  

命題変換(_2,_1,使役,[か,変,未然],[φ,下一,Kt]) :-  

    させる(_2,_1,Kt).  

命題変換(_2,_1,使役,[_3,下一,未然],[φ,下一,Kt]) :-  

    させる(_2,_1,Kt).  

命題変換(_2,_1,使役,[_3,上一,未然],[φ,下一,Kt]) :-  

    させる(_2,_1,Kt).  

%*****%
% アスペクト & テンス %
%*****%
% + 状態,+ 総統 %
%*****%
アスペクト(_2,_1,[+,-],[[],過去,[が,5,連用音便],[φ,φ,Kt]]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[+,-],[[],過去,[な,5,連用音便],[φ,φ,Kt]]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[+,-],[[],過去,[ば,5,連用音便],[φ,φ,Kt]]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[+,-],[[],過去,[ま,5,連用音便],[φ,φ,Kt]]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[+,-],[[],過去,[_4,_5,_3],[φ,φ,Kt]]) :-  

    (_3=連用音便;_3=連用),  

    た(_2,_1,Kt).  

アスペクト(_1,_1,[+,-],[[],[],Kt,Kt]).  

%*****%
% + 状態,- 総統 %
%*****%
アスペクト(_2,_1,[+,-],未完了,過去,[が,5,連用音便],[φ,φ,Kt]) :-  

    ていた(_2,_1,Kt).  

アスペクト(_2,_1,[+,-],未完了,過去,[な,5,連用音便],[φ,φ,Kt]) :-  

    ていた(_2,_1,Kt).  

アスペクト(_2,_1,[+,-],未完了,過去,[ば,5,連用音便],[φ,φ,Kt]) :-  

    ていた(_2,_1,Kt).  

アスペクト(_2,_1,[+,-],未完了,過去,[ま,5,連用音便],[φ,φ,Kt]) :-  

    ていた(_2,_1,Kt).  

アスペクト(_2,_1,[+,-],未完了,過去,[_4,_5,_3],[φ,φ,Kt]) :-  

    (_3=連用音便;_3=連用),  

    ていた(_2,_1,Kt).  

アスペクト(_2,_1,[+,-],未完了,[[],[が,5,連用音便],[φ,φ,Kt]]) :-  

    でいる(_2,_1,Kt).  

アスペクト(_2,_1,[+,-],未完了,[[],[な,5,連用音便],[φ,φ,Kt]]) :-  

    でいる(_2,_1,Kt).  

アスペクト(_2,_1,[+,-],未完了,[[],[ば,5,連用音便],[φ,φ,Kt]]) :-  

    でいる(_2,_1,Kt).  

アスペクト(_2,_1,[+,-],未完了,[[],[ま,5,連用音便],[φ,φ,Kt]]) :-  

    でいる(_2,_1,Kt).  

アスペクト(_2,_1,[+,-],未完了,[[],[_4,_5,_3],[φ,φ,Kt]]) :-  

    (_3=連用音便;_3=連用),  

    でいる(_2,_1,Kt).  

アスペクト(_1,_1,[+,-],[[],[],Kt,Kt]).  

%*****%
% - 状態,- 総統 %
%*****%
アスペクト(_2,_1,[+,-],未完了反復,過去,[_3,連用],[φ,Kt]) :-  

    つつあった(_2,_1,Kt).  

アスペクト(_2,_1,[+,-],未完了反復,[[],[_3,連用],[φ,Kt]]) :-  

    つつある(_2,_1,Kt).  

アスペクト(_2,_1,[+,-],未完了結果,過去,[が,5,連用音便],[φ,φ,Kt]) :-  

    ていた(_2,_1,Kt).  

アスペクト(_2,_1,[+,-],未完了結果,過去,[な,5,連用音便],[φ,φ,Kt]) :-  

    ていた(_2,_1,Kt).  

アスペクト(_2,_1,[+,-],未完了結果,過去,[ば,5,連用音便],[φ,φ,Kt]) :-  

    ていた(_2,_1,Kt).  

アスペクト(_2,_1,[+,-],未完了結果,過去,[ま,5,連用音便],[φ,φ,Kt]) :-  

    ていた(_2,_1,Kt).  

アスペクト(_2,_1,[+,-],未完了結果,過去,[_4,_5,_3],[φ,φ,Kt]) :-  

    (_3=連用音便;_3=連用),  

    ていた(_2,_1,Kt).

```

(次ページに続く)

```

アスペクト(_2,_1,[-, -],未完了結果,[],[が, 5,連用音便],[φ, φ, Kt]) :-  

    でいる(_2,_1,Kt).  

アスペクト(_2,_1,[-, -],未完了結果,[],[な, 5,連用音便],[φ, φ, Kt]) :-  

    でいる(_2,_1,Kt).  

アスペクト(_2,_1,[-, -],未完了結果,[],[ば, 5,連用音便],[φ, φ, Kt]) :-  

    でいる(_2,_1,Kt).  

アスペクト(_2,_1,[-, -],未完了結果,[],[ま, 5,連用音便],[φ, φ, Kt]) :-  

    でいる(_2,_1,Kt).  

アスペクト(_2,_1,[-, -],未完了結果,[],[_4,_5,_3],[φ, φ, Kt]) :-  

    (_3=連用音便;_3=連用),  

    でいる(_2,_1,Kt).  

アスペクト(_2,_1,[-, -],完了,過去,[が, 5,連用音便],[φ, φ, Kt]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-, -],完了,過去,[な, 5,連用音便],[φ, φ, Kt]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-, -],完了,過去,[ば, 5,連用音便],[φ, φ, Kt]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-, -],完了,過去,[ま, 5,連用音便],[φ, φ, Kt]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-, -],完了,過去,[_4,_5,_3],[φ, φ, Kt]) :-  

    (_3=連用音便;_3=連用),  

    た(_2,_1,Kt).  

アスペクト(_2,_1,[-, -],完了,[が, 5,連用音便],[φ, φ, Kt]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-, -],完了,[な, 5,連用音便],[φ, φ, Kt]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-, -],完了,[ば, 5,連用音便],[φ, φ, Kt]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-, -],完了,[ま, 5,連用音便],[φ, φ, Kt]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-, -],完了,[_4,_5,_3],[φ, φ, Kt]) :-  

    (_3=連用音便;_3=連用),  

    た(_2,_1,Kt).  

アスペクト(_2,_1,[-, -],[],過去,[が, 5,連用音便],[φ, φ, Kt]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-, -],[],過去,[な, 5,連用音便],[φ, φ, Kt]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-, -],[],過去,[ば, 5,連用音便],[φ, φ, Kt]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-, -],[],過去,[ま, 5,連用音便],[φ, φ, Kt]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-, -],[],過去,[_4,_5,_3],[φ, φ, Kt]) :-  

    (_3=連用音便;_3=連用),  

    た(_2,_1,Kt).  

アスペクト(_1,_1,[-, -],[],[],Kt,Kt).  

%*****  

% - 状態, + 繼続 %  

%*****  

アスペクト(_2,_1,[-, +],未完了,過去,[が, 5,連用音便],[φ, φ, Kt]) :-  

    でいた(_2,_1,Kt).  

アスペクト(_2,_1,[-, +],未完了,過去,[な, 5,連用音便],[φ, φ, Kt]) :-  

    でいた(_2,_1,Kt).  

アスペクト(_2,_1,[-, +],未完了,過去,[ば, 5,連用音便],[φ, φ, Kt]) :-  

    でいた(_2,_1,Kt).  

アスペクト(_2,_1,[-, +],未完了,過去,[ま, 5,連用音便],[φ, φ, Kt]) :-  

    でいた(_2,_1,Kt).  

アスペクト(_2,_1,[-, +],未完了,過去,[_4,_5,_3],[φ, φ, Kt]) :-  

    (_3=連用音便;_3=連用),  

    でいた(_2,_1,Kt).  

アスペクト(_2,_1,[-, +],未完了,[が, 5,連用音便],[φ, φ, Kt]) :-  

    でいる(_2,_1,Kt).  

アスペクト(_2,_1,[-, +],未完了,[な, 5,連用音便],[φ, φ, Kt]) :-  

    でいる(_2,_1,Kt).  

アスペクト(_2,_1,[-, +],未完了,[ば, 5,連用音便],[φ, φ, Kt]) :-  

    でいる(_2,_1,Kt).  

アスペクト(_2,_1,[-, +],未完了,[ま, 5,連用音便],[φ, φ, Kt]) :-  

    でいる(_2,_1,Kt).  

アスペクト(_2,_1,[-, +],未完了,[_4,_5,_3],[φ, φ, Kt]) :-  

    (_3=連用音便;_3=連用),  

    でいる(_2,_1,Kt).  

アスペクト(_2,_1,[-, +],完了,過去,[が, 5,連用音便],[φ, φ, Kt]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-, +],完了,過去,[な, 5,連用音便],[φ, φ, Kt]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-, +],完了,過去,[ば, 5,連用音便],[φ, φ, Kt]) :-  

    だ(_2,_1,Kt).

```

(次ページに続く)

```

アスペクト(_2,_1,[-,+],完了,過去,[ま,5,連用音便],[φ,φ,Kt]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-,+],完了,過去,[_4,_5,_3],[φ,φ,Kt]) :-  

    (_3=連用音便;_3=連用),  

    た(_2,_1,Kt).  

アスペクト(_2,_1,[-,+],完了,[[],[が,5,連用音便],[φ,φ,Kt]]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-,+],完了,[[],[な,5,連用音便],[φ,φ,Kt]]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-,+],完了,[[],[ば,5,連用音便],[φ,φ,Kt]]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-,+],完了,[[],[ま,5,連用音便],[φ,φ,Kt]]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-,+],完了,[[],[_4,_5,_3],[φ,φ,Kt]]) :-  

    (_3=連用音便;_3=連用),  

    た(_2,_1,Kt).  

アスペクト(_2,_1,[-,+],[],過去,[が,5,連用音便],[φ,φ,Kt]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-,+],[],過去,[な,5,連用音便],[φ,φ,Kt]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-,+],[],過去,[ば,5,連用音便],[φ,φ,Kt]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-,+],[],過去,[ま,5,連用音便],[φ,φ,Kt]) :-  

    だ(_2,_1,Kt).  

アスペクト(_2,_1,[-,+],[],過去,[_4,_5,_3],[φ,φ,Kt]) :-  

    (_3=連用音便;_3=連用),  

    た(_2,_1,Kt).  

アスペクト(_1,_1,[-,+],[],[],Kt,Kt).

```

%*****アスペクト活用 %

```

%*****た %  

た([-32067,-32021!_1],_1,未然).  

た(_1,_1,連用).  

た([-32067!_1],_1,終止).  

た([-32067!_1],_1,連体).  

%*****だ %  

だ([-32066,-32021!_1],_1,未然).  

だ(_1,_1,連用).  

だ([-32066!_1],_1,終止).  

だ([-32066!_1],_1,連体).  

%*****でいた %  

でいた([-32059,-32094!_1],_1,未然).  

でいた([-32059,-32094!_1],_1,連用).  

でいた([-32059,-32094,-32067!_1],_1,終止).  

でいた([-32059,-32094,-32067!_1],_1,連体).  

%*****ていた %  

ていた([-32060,-32094!_1],_1,未然).  

ていた([-32060,-32094!_1],_1,連用).  

ていた([-32060,-32094,-32067!_1],_1,終止).  

ていた([-32060,-32094,-32067!_1],_1,連体).  

%*****でいる %  

でいる([-32059,-32094!_1],_1,未然).  

でいる([-32059,-32094!_1],_1,連用).  

でいる([-32059,-32094,-32023!_1],_1,終止).  

でいる([-32059,-32094,-32023!_1],_1,連体).  

%*****ている %  

ている([-32060,-32094!_1],_1,未然).  

ている([-32060,-32094!_1],_1,連用).  

ている([-32060,-32094,-32023!_1],_1,終止).  

ている([-32060,-32094,-32023!_1],_1,連体).

```

リスト1(続き)

```

%%%%%
% つつあった %
%%%%%
つつあった([-32062,-32062!_1],_1,未然).
つつあった([-32062,-32062,-32096,-32024!_1],_1,連用音便).
つつあった([-32062,-32062,-32096,-32063!_1],_1,連用音便).
つつあった([-32062,-32062,-32096,-32063,-32067!_1],_1,終止).
つつあった([-32062,-32062,-32096,-32063,-32067!_1],_1,連体).

%%%%%
% つつある %
%%%%%
つつある([-32062,-32062!_1],_1,未然).
つつある([-32062,-32062,-32096,-32024!_1],_1,連用).
つつある([-32062,-32062,-32096,-32063!_1],_1,連用音便).
つつある([-32062,-32062,-32096,-32023!_1],_1,終止).
つつある([-32062,-32062,-32096,-32023!_1],_1,連体).

%%%%%
% 話者の判断 %
%%%%%
[%] %

話者の判断(_1,_1,[[],Kt,Kt]).
話者の判断(_2,_1,打ち消し,[_3,_4,未然],[φ,φ,Kt]) :-  

    ない(_2,_1,Kt).
話者の判断(_2,_1,希望,[_3,_4,連用],[φ,φ,Kt]) :-  

    たい(_2,_1,Kt).
話者の判断(_2,_1,推定,[_3,_4,終止],[φ,φ,Kt]) :-  

    らしい(_2,_1,Kt).
話者の判断(_2,_1,様態,[_3,_4,連用],[φ,φ,Kt]) :-  

    そうだ(_2,_1,Kt).
話者の判断(_2,_1,伝聞,[_3,_4,終止],[φ,φ,Kt]) :-  

    そうだ(_2,_1,Kt).
話者の判断(_2,_1,不確定,[_3,_4,連体],[φ,φ,Kt]) :-  

    ようだ(_2,_1,Kt).
話者の判断(_2,_1,丁寧,[_3,_4,連用],[φ,φ,Kt]) :-  

    ます(_2,_1,Kt).
話者の判断(_2,_1,意志,[_3,_5,未然],[φ,φ,Kt]) :-  

    う(_2,_1,Kt).
話者の判断(_2,_1,意志,[_3,_4,未然],[φ,φ,Kt]) :-  

    よう(_2,_1,Kt).
話者の判断(_2,_1,打ち消し意志,[_3,_5,終止],[φ,φ,Kt]) :-  

    まい(_2,_1,Kt).
話者の判断(_2,_1,打ち消し意志,[_3,_4,未然],[φ,φ,Kt]) :-  

    まい(_2,_1,Kt).

%%%%%
% 判断時制 %
%%%%%
判断時制(_2,_1,過去,_3,Kt) :-  

    (_3=連用音便;_3=連用),
    た(_2,_1,Kt).
判断時制(_1,_1,[[],Kt,Kt]).

%%%%%
% 話者の働きかけ %
%%%%%
話者の働きかけ([-32030,-32025!_3],_3,連言,[_1,_2,連体]).  

話者の働きかけ([-32087!_3],_3,連言,[_1,_2,終止]).  

話者の働きかけ([-32056,-32057!_3],_3,例示,[_1,_2,連体]).  

話者の働きかけ([-32083,-32025,-32094!_3],_3,程度,[_1,_2,連体]).  

話者の働きかけ([-32082,-32025,-32094!_3],_3,程度,[_1,_2,連体]).  

話者の働きかけ([-32036,-32059!_3],_3,帰着点,[_1,_2,連体]).  

話者の働きかけ([-32077,-32090!_3],_3,類推,[_1,_2,連用]).  

話者の働きかけ([-32059,-32032!_3],_3,類推,[_1,_2,連用]).  

話者の働きかけ([-32066,-32063,-32060!_3],_3,類推,[_1,_2,連用]).  

話者の働きかけ([-32075,-32087!_2],_2,限定,[形容詞,_1,連用]).  

話者の働きかけ([-32075,-32087!_2],_2,限定,[形容動詞,_1,連用]).  

話者の働きかけ([-32075,-32087!_3],_3,限定,[_1,_2,連体]).  

話者の働きかけ([-32050,-32087,-32024!_3],_3,限定,[_1,_2,連体]).  

話者の働きかけ([-32066,-32081!_3],_3,限定,[_1,_2,連体]).  

話者の働きかけ([-32050!_3],_3,仮定,[_1,_2,仮定]).  

話者の働きかけ([-32058!_3],_3,仮定,[_1,_2,終止]).
```

(次ページに続く)

```

話者の働きかけ([-32087,-320251_3],_3,原因,[_1,_2,終止]).
話者の働きかけ([-32052,-320591_3],_3,原因,[_1,_2,連体]).
話者の働きかけ([-320591_1],_1,順接,[が,5,連用]).
話者の働きかけ([-320591_1],_1,順接,[な,5,連用]).
話者の働きかけ([-320591_1],_1,順接,[ば,5,連用]).
話者の働きかけ([-320601_3],_3,順接,[_1,_2,連用]).
話者の働きかけ([-32066,-320241_1],_1,並列,[が,5,連用]).
話者の働きかけ([-32066,-320241_1],_1,並列,[な,5,連用]).
話者の働きかけ([-32066,-320241_1],_1,並列,[ば,5,連用]).
話者の働きかけ([-32066,-320241_1],_1,並列,[ま,5,連用]).
話者の働きかけ([-32067,-320241_1],_1,並列,[_1,_2,連用]).
話者の働きかけ([-32056,-32086,-320251_2],_2,並列,[形容詞,_1,終止]).
話者の働きかけ([-32056,-32086,-320251_3],_3,並列,[_1,_2,連用]).
話者の働きかけ([-32059,-320321_1],_1,逆接,[が,5,連用]).
話者の働きかけ([-32059,-320321_1],_1,逆接,[な,5,連用]).
話者の働きかけ([-32059,-320321_1],_1,逆接,[ば,5,連用]).
話者の働きかけ([-32060,-320321_3],_3,逆接,[_1,_2,連用]).
話者の働きかけ([-32081,-32022,-32057,-320321_3],_3,逆接,[_1,_2,終止]).
話者の働きかけ([-32081_3],_3,逆接,[_1,_2,終止]).
話者の働きかけ([-32052,-320551_3],_3,逆接,[_1,_2,連体]).
話者の働きかけ([3,_3,命令,[_1,_2,命令]]).
話者の働きかけ([-320601_4],_4,命令,[_2,_3,_1]) :-  

    (_1=連用音便; _1=連用).
話者の働きかけ([-32056,-32077,-320941_4],_4,命令,[_2,_3,_1]) :-  

    (_1=連用音便; _1=連用).
話者の働きかけ([-320871_3],_3,疑問,[_1,_2,連体]).
話者の働きかけ([-320521_3],_3,疑問,[_1,_2,連体]).
話者の働きかけ([-320561_3],_3,禁止,[_1,_2,連体]).
話者の働きかけ([-320531_3],_3,確認,[_1,_2,終止]).
話者の働きかけ([3,_3,平叙,[_1,_2,終止]]).

```

リスト 1 (続き)

```

%*****%
% 生成時 %
% 動詞命題の処理 %
%*****%
述語生成([[S0,S1],[_行,_段],,_Asp],_命題,Osr,T1,Mode,T2,Func) :-  

    命題生成(S0,[[],[S0,S1],[_行,_段],,_Asp],_命題,Osr,T1,Mode,T2,Func),  

    knjprint(S0).
命題生成(S0,St,[[S0,S1],[_行,_段],,_Asp],_命題,Osr,T1,Mode,T2,Func) :-  

    M6 =.. [話者の働きかけ,S6,St,Func,Kt5],  

    M6,  

    M5 =.. [判断時制,S5,S6,T2,Kt4,Kt5],  

    M5,  

    M4 =.. [話者の判断,S4,S5,Mode,Kt3,Kt4],  

    M4,  

    M3 =.. [アスペクト,S3,S4,Asp,Osr,T1,Kt2,Kt3],  

    M3,  

    M2 =.. [命題変換,S2,S3,_命題,[_行,_段,Kt1],Kt2],  

    M2,  

    活用形(_行,_段,活用),  

    M1 =.. [_活用,S1,S2,Kt1],  

    M1.  

活用形(_行,_段,活用) :-  

    name(_行,N1),  

    name(_段,N2),  

    append(N1,N2,N3),  

    name(_活用,N3),  

    !.  

append([A:X],Y,[A:Z]) :-  

    append(X,Y,Z).
append([],X,X).
knjprint([]).
knjprint([X|Y]) :-  

    A is X / 256 + 255,  

    B is X mod 256 + 256,  

    put(A),put(B),
    knjprint(Y).

```

リスト 2　述語生成ルーチンのプログラム