マイクロ・コンピューターによる自動品詞認定の試み

西端幸雄

私は、以前本誌21号において「マイコンによる索引作り」という報告書を発表した。また、それと相前後して、幾人かの方々が同種の考え、試みを発表された。ただ、私の考え方も含め、それらの報告書を見るに、マイコンに索引作りをさせる過程で、人間の介在する箇所が多過ぎるように思えてならなかった。その一番の問題点は、品詞認定という作業過程である。この品詞認定は、扱う文献資料の量が多くなればなるほど、一人の手で処理するのが難しくなる。そのために、どうしても何人かに、それも学生に分担して行なわせることになる。すると、ここで幾多の問題が生じる。そのひとつは、しっかりした品詞認定の基準を作っておいても、各人の処理結果におのずとズレが生じる点、また、各人の処理能力に差があるため、作業が遅延しやすい点、さらに、この点が最も大きな問題点であるが、今日の学生（少なくとも、私の周囲にいる学生）の文法についての知識・能力は、かなり低い点等々である。

そこで、今回、考えたのは、文章を語句単位に区切る所までは人間が行ない、以後品詞認定から50音順に並べ替え、印刷するまでを、すべてマイコンに行なわせるというプログラムである。ただし、本報告は、その中の品詞認定の段階に限っておく。その理由は、この品詞認定の段階が今回の作業の中心であるという点と、後述するように、今回の作業過程を全て詳述するとなると、6種類のプログラムを紹介しなければならなくなるが、紙幅の関係でとても無理である点などである。

ところで、世間では、コンピューターに何かの作業を行なわせれば、何でも完璧にこなしてくれるように思いがちだが、それは、幻想にしかすぎない。コンピューターといえども、人間の作ったプログラムがなければ、全くの無用の長物でしかない。換言すれば、人間の作ったプログラムの良否がコンピューターの作業能力の良否を決定すると言えよう。よって、今回のプログラム作成にあたっては、なるべく完璧なものを目差しながら、とりあえず、品詞認定の正答率90〜95パーセント以上を当面の目標とした。まだ不十分な点が多々残っているものの、一応その目標を達成できたので、ここに中間報告として、以下に記す。

1. 使用機種

今回の作業で用いたマイコンは、FUJITSU MICRO 16β（以下、FM16β と略す。）の HD タイプである。この機種を用いたのは、発売当時（昭和59年11月）、16ビットのマイコンとしては、唯一メインメモリが 512KByte あり、また、JIS 第2水準までの漢字（計6,349字）を標準実装している点と、漢字・平仮名をワープロ風に簡便に打ち出せる点、さらには、処理速度が早い点などからである。また、HD タイプというのは、10メガバイトのハードディスクと1メガバイトのミニフロッピィディスクを1ドライブ備えているものである。マイコンにおいては、漢字・平仮名等を用いる場合、2バイト必要となる。ということは、漢字・平仮名を記憶させる際、10メガバイトのハードディスクでは、単純に言えば、500万文字、1メガのミニフロッピィディスクでは、50万文字が記憶可能ということになる。後述するように、ディスク上に辞書を登録し、使用する際には、どうしても、以上に述べた程度のディスクの容量が必要となろう。また、大量の言語データを処理するには、前述の各種機能も絶対的に必要となろう。

2. 各種プログラム

今回の作業のために作成したプログラムは、以下の6種類である。

- ○辞書作成
- ○データ登録
- ○品詞認定
- ○品詞決定・漢字情報付加
- ○50音順並べ替え（未完）
- ○印刷（未完）

「辞書作成」プログラムは、品詞を決定し、漢字情報を付加する際に用いる辞書をディスク上に作成するものである。旺文社刊『古語辞典』に掲載されている見出し語の中から約3万語を、このプログラムによって登録した。登録に際しては、まず、語句を短単位（5文字以下）と長単位（6文字以上）に分け、それぞれを別々に、アからヲまで47種の頭音別に分割登録した。さらに、付属情報として、漢字情報、用言には活用行と活用の種類の情報を付けた。また、同音語のある場合は、『古典対照語い表』〈宮島達夫氏編〉等も参考にしながら、使用頻度の高いものから順に登録するように心掛けた。

「データ登録」プログラムは、品詞認定したい文章データを事前にディスクに登録するものである。その際、事前の作業として、前述したように、各語句の区切り目に「｜」記号を、また、複合語の区切り目に「＝」記号を付けておき、それをそのまま

登録するようにした。また、韻文、散文の別なく、複数行にわたっているものも登録できるように配慮した。

「品詞決定・漢字情報付加」プログラムは、「品詞認定」プログラムの認定結果の確認と漢字情報を付ける作業を、辞書データを介して行ない、最終決定された語句データをディスクに登録するものである。各語句の中、特に用言については、終止形に変換し、辞書データと照合することにより、認定の正誤判断および訂正を自動的に行なわせるように心掛けた。また、語句データの登録は、現在のところ、そのまま登録する形式をとっているが、後日の使用の便を考え、様々な情報（例えば、接続の様態を知る情報とか特定の品詞だけを取り出す情報）を同時に登録するような方法を今後検討してゆきたいと思う。

3. 品詞認定

「品詞認定」プログラムを作成するにあたり、最も参考になったのは、人間が品詞認定を行なう際に踏む手順である。この認定過程における人間の思考・行動を後辿ることにより、プログラム上でも、より正確な認定が行なえると考えたわけである。

では、人間が品詞認定を行なう際、どのような思考・行動をしているだろうか。それを、大別すると、以下の３つの思考・行動をしていると言えよう。

① 辞書・参考書による判断

人間は、品詞認定をしようとする時、よく文法参考書や辞書に手掛りを求める。そして、当該の語句と掲載事項を照合して、判断を行なう。この過程を取り入れたのが、「品詞認定」および「品詞決定・漢字情報付加」プログラムである。しかし、この過程の中でも辞書による判断を先行させなかった理由は、ひとつには、辞書データと照合させて、品詞認定をしたとしても、最終決定は、後述の語形・接続による判断をどうしても経なければならない。それよりは、語形・接続による判断だけで、かなりの正答率を得ておいた方が時間的には短くてすむという点と、富士通のディスクの立ち上げ速度が遅く、検索時間が１語につき、ハードディスクで約１秒、ミニフロッピィディスクでは約３秒もかかる。すると、仮に15語を検索すると、それだけで15秒ないしは45秒もかかってしまうという点からである。ただ、後述するように、語形による判断を下すのに必要な辞書的データを本プログラム内に付随させて用いた。

② 語形による判断

この語形による判断を下す際、人間は、語句の３点の語形に目を向けているのではないだろうか。その１点は、語頭である。例えば、語頭２文字が「うち」であれば、動詞の可能性があるとか。２点目は、語尾である。例えば、語尾３文字が「しから、しかり、しかる、しかれ」とあれば、シク活用の形容詞である可能性が大きいとか。３点目は、全体である。例えば、語尾１文字が同じ「み」であっても、「いづみ」と「あらみ」と「うらみ」とでは、それぞれ文法的性格は異なっているとか。その他、語形についての知識を豊富に持っている者は、例えば、語尾１文字が「おござほぼゃ

ゅょわん」のいずれかであれば、名詞の可能性が大きいと判断するであろう。本プログラムでは、こうした語形による判断の過程を重視し、以下に掲げたデータを後に説明する「DATASUB」プログラム内に「DATA」文として収納した。

ⓐ『古典対照語い表』を手掛りに、使用頻度の高い語句を各品詞毎に収納。

ⓑ 助詞、助動詞についても、すべて収納。特に助動詞は、各語毎に活用の全てを収納。

ⓒ『日本語尾音索引―古語篇―』〈丹羽一彌氏・田島毓堂氏共編〉、『古典対照語い表』を手掛りに検出した、体言の絶対的条件となる語頭2文字と3文字、語尾2文字と3文字を収納。同じく、両書によって検出した、動詞の絶対的条件となる語尾2文字を収納。

③ 接続による判断

品詞認定の過程で、人間が最も意を用いるのは、この接続による判断である。特に助詞、助動詞を手掛りに、上接の品詞や用言の活用形や活用の種類を決定しようとする。例えば、打消の助動詞「ず」には、活用語の未然形が上接するとか、名詞には、活用形の連体形が主に上接するとか。ただ、この接続による判断を下す際、人間は、接続関係だけで品詞認定を行なうこともあるが、多くの場合、また、より厳密な判断を下そうとする場合は、語形による判断をも加味して、品詞認定をしようとしているのではないだろうか。例えば、打消の助動詞「ず」に、語尾がア段音の語句が上接していれば、その語句を4段活用の動詞の未然形と判断するであろう。本プログラムでも、この接続による判断のステップが全体の約半分を占めている。そのほとんどは、助詞、助動詞を手掛りに、上接する語句の品詞や活用形、活用の種類を決定しようとするものである。その際、人間が行なうと同様に、上接語の語形、特に語尾音を考慮するという語形による判断をも加味させ、より正確な認定を行なえるように心掛けた。

以上の3つの思考・行動の他に、構文関係を把えて、人間は、品詞認定することも多い。しかし、これをプログラム化すると、非常に複雑なものになってしまうので、今回は、極力、この過程は省略した。ただ、その中の若干のものは、必要上、プログラム内に取り入れた。例えば、文頭には助詞、助動詞が絶対に立たないということや文中に係助詞があれば、近接した所か文末に連体形が立つといったような点である。

このような、品詞認定における人間の思考・行動を忠実にプログラム上に反映させ、さらに若干の補正過程を加えることにより、より精度の高いものを目指した。

4. プログラム

「品詞認定」プログラムは、大別して、2つのプログラムからなる。それぞれ「MAIN」プログラムと「DATASUB」プログラムである。「MAIN」プログラムは、正に品詞認定を行なうものであり、448ステップからなっている。「DATASUB」プログラムは、前項の②（語形による判断）で述べた、主として語形による判断を下すの

に必要な各種データを「READ…DATA」文により配列変数内に読み込み、「MAIN」プログラムに引き渡すものである。このように、プログラムを2つに分割せざるを得なかったのは、「MAIN」プログラムが長大になり、2つのプログラムを合せると、メモリのテキスト領域（プログラムを記憶し、作動させるメモリ領域）を超えてしまうからである。こうした制約を克服するために、現在の BASIC 言語には、2つのプログラム間でデータを受け渡しする命令「CHAIN」「COMMON」が用意されている。「DATASUB」プログラムでは、この中の「CHAIN」命令を用い、配列変数内に読み込んだ、全てのデータを「MAIN」プログラムに引き渡せるようにしている。

以上の2つのプログラムの中、「DATASUB」プログラムは、いわば辞書的性格を持ったものだといえる。基本的には「DIM」文によって宣言した各種の配列変数に、「READ…DATA」文によって、データを読み込み、前述の「CHAIN」命令で「MAIN」プログラムに配列変数のまま、データを引き渡すという作業を行なうだけである。ただ、このプログラム内に収納されるデータの量と質が「MAIN」プログラムによる品詞認定の精度をある程度左右するという点では、非常に大きな存在価値がある。また、FM16β では、「MAIN」プログラム起動時の配列変数領域（配列変数を記憶するメモリ領域）にかなりの余裕があるので、この「DATASUB」プログラム内のデータを増補する余地が十分にある。そのことは、ひいては品詞認定の精度を高めることにもつながる。

次に、「MAIN」プログラムであるが、これは、細分すると、おおよそ下記の8種の区画（サブルーチン）からなる。後の補正を考慮し、プログラムを組み立てるに際しては、できる限り構造化を図った。

ⓐ 初期処理・メニュー
ⓑ 文章データの読み込み
ⓒ 文章データの区切り
ⓓ 語形による判断
ⓔ 同音異義語の判別
ⓕ 非活用語の判別
ⓖ 活用語の判別
ⓗ 接続による判断

「MAIN」プログラム全体の流れは、基本的には、ⓐからⓗへと各ルーチンを経ている。ただ、できる限り不要なルーチンを経ないように、ⓓ（語形による判断）、ⓗ（接続による判断）の各ルーチン内に語句データの文法的性格付けをする情報をプログラム化した。その点について、詳しくは、後述する。以下、各ルーチンの説明をしておく。

ⓐ 初期処理・メニュー

このルーチンは、「データ登録」プログラムによって、すでに記録されている文章データの種類をディスプレイ上に表示し、使用者が品詞認定したい文章データのファイルを、指示によりオープンするものである。

当面の作業として、八代集の和歌、詞書、左注、序文、作者名を処理する予定なので、最初の画面には、八代集の作品名を表示し、指示を待つ。指示があれば、当該の作品中の文章データの種類（和歌、詞書等）を表示し、指示を待つ。指示があれば、当該のファイルをオープンして、次のルーチンへ進むという流れを経る。以上の説明からもわかるように、このルーチンは、作業開始時に使用するだけであるので、常時「MAIN」プログラム上に置く必要があるのか、只今検討しているところである。ただ、仮に複数の種類の異なった文章データを連続処理する場合には、「MAIN」プログラム上に置いておく方が便利ではある。

ⓑ 文章データの読み込み

このルーチンは、まず、ⓐ（初期処理・メニュー）で指示された作品の当該の文章データを全て処理するのか、一部を処理するのか、一部を処理するならどの範囲なのかといった内容を表示し、指示を待つ。指示があれば、指示された範囲の文章データをディスクからマイコンへ順次読み込み、次のルーチンへ引き渡すという流れを経る。

ⓒ 文章データの区切り

このルーチンは、「データ登録」プログラムによって、ディスクへ登録された区切り符号付きの文章データが、ⓑ（文章データの読み込み）ルーチンにより、1文（1単位）ずつマイコンへ読み込まれた後、その区切り符号（｜）を目じるしに、区切り符号の直前までを1語句として、配列変数に語句データを収納するという流れを経る。また、複合語についても、その区切り符号（＝）を目じるしに、上接要素と下接要素に分割し、必要に応じて、それぞれ文字変数に代入した。

ⓓ 語形による判断

このルーチン以下が実質的な品詞認定を行なっているプログラムである。その中で、このルーチンは、語形によって、品詞が完全に決定するものには品詞コードを付ける。そうでないものについては、おおよその目安を付け、可能性のある品詞を認定するサブルーチンへ引き渡すという流れを経る。

本プログラムにおける語形による判断の方法は、語句の文字数と、語尾音、または語頭音と語尾音の関係で判断を加えた。例えば、文字数に関係なく、語尾が「おござほぼゃゅょわん」の語句は、「いざ」「なほ」を除き、名詞となる可能性が高い。また、文字数が2文字以上で、語頭が「あいうえおきぎくぐげじずぜちぢづでねひびふぶほぼみゆりろわゐゑ」のものや、語尾が「あいうえぎぐけげこさざぜただちぢづぬねばひびふぶべまゆろゐゑ」のものは、付属語ではありえないので、自立語と判断し、「非活用語の判別」ルーチンから「活用語の判別」ルーチンへと流れるようにした。また、文字数が2文字以上で、語頭が「かがこさだつてとどなにのばまもよを」であって、語尾が「かがしそぞつててどなにのはへみむもやよらりを」のものは、助詞の可能性が高いので、まず助詞を判別するルーチンへゆき、データが合致すれば、品詞コードを付け、合致しなければ、「非活用語の判別」ルーチンから「活用語の判別」ルーチンへ流れるようにした。

このルーチンは、あくまでプログラムの無駄な流れを防ぐということと一応の目安を付けるということを第一義に考え、この段階では、品詞を認定することは極力ひかえた。よって、語句データを次の4種に分類するにとどめた。

- 名詞の可能性のあるもの
- 自立語　〃
- 助詞　〃
- 助動詞　〃

また、語形による判断は、このルーチン以外でも、プログラムの随所で行なった。その中で、最もよく用いた方法は、語尾音による判断である。特に、動詞の活用形や活用の種類を判断しようとする際、この語形による判断が有効な働きをする。その方法は、語尾を文字そのもので把えたなら、プログラムが長くなり、処理時間がかかるので、語尾の文字をアスキーコードで表わし、以下に述べるような計算を行なうというものである。「ア」から「ン」までのアスキーコードは、以下の通りである。

ア	カ	サ	タ	ナ	ハ	マ	ヤ	ラ	ワ
177	182	187	192	197	202	207	212	215	220
イ	キ	シ	チ	ニ	ヒ	ミ		リ	ヲ
178	183	188	193	198	203	208		216	166
ウ	ク	ス	ツ	ヌ	フ	ム	ユ	ル	ン
179	184	189	194	199	204	209	213	217	221
エ	ケ	セ	テ	ネ	ヘ	メ		レ	
180	185	190	195	200	205	210		218	
オ	コ	ソ	ト	ノ	ホ	モ	ヨ	ロ	
181	186	191	196	201	206	211	214	219	

この表を見るとわかるように、ア行からマ行までの各段は、それぞれ「177、178、179、180、181」を元として、5ずつ数値が増加している。ということは、アスキーコードを5で割った余りを求めれば、アからオの各段が求められることになる。つまり、次の式によって、語尾がどの段の音であるかがわかる。

（語尾のアスキーコード）MOD 5 =X

〔「MOD」命令は商の余りを求める〕

Xが2であれば、ア段音、3であれば、イ段音、4であれば、ウ段音、0であれば、エ段音、1であれば、オ段音となる。これを動詞の活用形や活用の種類を求める場合に応用すると、例えば、Xが2であるものは、4段活用の未然形であるとか、下接する語句が連用形接続のものであって、Xが0であり、アスキーコードが183（ケ）でなければ、下2段活用であるといった判断が下せる。

さらに、3．品詞認定の項でも若干述べたが、「非活用語の判別」ルーチンと「活用語の判別」ルーチンも、いわば、この語形による判断と同種の作業を行なってい

る。ただ、この場合は、データが大量になるため、「DATASUB」プログラムにおいて、配列変数内に読み込んでおいたデータと語句データを「SEARCH」命令を用いて照合し、合致すれば、品詞コードを付けるという方法を取っている。この「SEARCH」命令は、下記のような式によって成り立つため、データ数の多い配列変数と照合する場合、「DATASUB」プログラムの「DATA」文内の各データ位置を各語頭音別に調べ、整理しておくと、語頭音を求めるたびに、検索開始位置のパラメータを変えるようにしておくことによって、無駄な検索を行なわなくてすむ。

SEARCH（配列変数名，検索データ，検索開始位置）＝検索位置

〔検索データが見付からない場合は、－1を返してくる。〕

また、上記の式でもわかるように、「SEARCH」命令で求められる数値が配列変数内における検索データの検索位置を答えとして返してくるため、活用語のデータとの照合については、次のような応用ができる。つまり、用言の場合、各語共に1語句につき、未然形から命令形までの6種の語形と漢字情報の計7データを「DATA」文内に収納しておく。そうすると、何語かをデータとして、配列変数内に読み込んでおいても、検索位置を7で割った余りによって、また、助動詞の場合も同様に、検索位置を6で割った余りによって各活用形が求められる。例えば、「DATASUB」プログラムの用言データが10語あるとすれば、各活用形と漢字情報を含めて、延70データが配列変数内に読み込まれている。そして、「SEARCH」命令によって、ある語句データが61番目のデータと合致したとすると、（61 MOD 7＝5）という式が成り立ち、活用形の5番目、つまり、その語句データの活用形は、已然形であることがわかる。

ⓔ 同音異義語の判別

日本語の場合、同音異義語が非常に多く、その全てについて、プログラム内において判別することは不可能である。また、このルーチンを整備、補強することがプログラムの精度を高めることになるのだが、現在「MAIN」プログラムだけでテキスト領域の9割ほどを使用しているため、これ以上プログラムを長くできない状態にもある。そのため、現在のところ、上接の語句の品詞認定に大きく貢献する助詞、助動詞で同音異義語となるものに限っている。「しすせてなぬねるれに」といったものがそれである。そして、これらが助詞であるか、助動詞であるかは、主として後述する接続による判断によって判別し、それが不可能な場合は、前述の語形による判断を加味した。接続による判断については後述するとして、語形による判断の方法は前項で述べたと同じように、上接の語句の語尾音のアスキーコードを5で割り、その余りを求め、語尾が何段の音を示しているかを探るというものである。そうすることにより、当該の「しすせてなぬねるれに」が助詞か助動詞であるか、また、上接する語句の品詞が何であるかがある程度明らかになる。例えば、「に」に上接する語句の語尾音がア段音かオ段音であれば、「に」は、助詞であり、上接する語句は、名詞であることがわかる。

ただ、上記の語句についてだけ、同音異義語を判別するにとどめると、精度が低くなるので、使用頻度の高い語句で、同音異義語を持つ「なく」「なる」「か」等につい

ては、多少構文的解釈を加味しながら判別を行なうようにした。例えば、「なく」の場合、「鳴く」「無く」の判別は、文章データ中に「うぐひす」「とり」「ほととぎす」等といった語句が存在すれば、4段活用の動詞「鳴く」だというように判断させた。

ⓕ 非活用語の判別

ⓖ 活用語の判別

これらのルーチンは、これまでに述べた「SEARCH」命令と「MOD」命令の繰り返しであるので、説明を省略する。

ⓗ 接続による判断

このルーチンは、前述したように、主として助詞、助動詞を手掛りに上接する語句の品詞や活用形、活用の種類を認定しようとするものである。そして、「品詞認定」プログラム内においては、このルーチンが品詞を最終的に認定するという役割を果す点で、非常に重要な部分である。

人間が接続によって、品詞を認定する際、一般的には下接する語句の品詞および接続の様態から上接する語句の品詞や活用形、活用の種類を判断しようとする。よって、本プログラムでも文章データを語句単位に区切り、配列変数内に読み込んだ後は、文末の語句データから文頭のものへという順序で、品詞認定してゆく方法を取り入れた。ただ、こうした逆順では認定できない場合、例えば、係結びの関係を探る場合などは、正順の方が認定しやすいので、適宜正順による認定方法をも取り入れた。

そして、このルーチンを有効に働かせ、無駄な流れのないプログラムにするため、特に助詞、助動詞については、接続の様態によって分類し、各語句に接続情報を付けた。分類の結果、助詞には、33種類、助動詞には、15種類の接続パターンがあることがわかった。それにより、「DATASUB」プログラムに収納した助詞、助動詞の全てにパターン毎の接続情報（数字やローマ字を1文字）を付け、また、このルーチンを各接続パターン毎に下位分割し、当該の接続パターンを判断するプログラム内だけを流れるようにした。例えば、助詞「の」に接続情報「A」を付けておくとすると、下接する語句が助詞「の」であり、接続情報として「A」が付いていれば、助詞「の」の接続による判断ルーチンにゆき、上接する語句の品詞認定を行なうという流れを経る。また、その他、下接する語句が名詞、用言および品詞不明の場合についても、接続による判断によって、上接する語句のおおよその品詞認定を行なうようにした。例えば、下接する語句が名詞であり、上接する語句の語尾2文字が「しく」となっていれば、上接する語句は、シク活用の形容詞の未然形ではなく、連用形の可能性が大きいというように判断させた。

○

以上のような流れを経て、各文章データの各語句を品詞認定した後、現在のところ、下に示したような形式で、ディスプレイ上に表示するか、プリンターによって印字するかしている。ただ、このような方法を取っているのは、現在、まだテストを繰

り返している段階であるからであって、実際に使用する場合は、１文章データ毎に各語句データと品詞コードデータ、および漢字や接続情報のあるものは、そうしたものも含め、一度ディスク上に記憶させ、次の「品詞決定・漢字情報付加」プログラムに引き渡すという方法を取る。

```
0080          TIME=00:00:04
  1 わが          02
  2 やど          00
  3 の            06
  4 かきね        00
  5 や            06
  6 はる          00
  7 を            06
  8 へだつ        D3
  9 らむ          94
 10 なつ          00
 11 き            カ2
 12 に            92
 13 けり          93
 14 と            06
 15 みゆる        54
 16 う＝の＝はな  00
```

5. 正答率・処理時間

今回、本プログラムを組むにあたって、テスト使用する文章データとして、『拾遺和歌集』の80番までの和歌と『小倉百人一首』の50番までの和歌を用いた。主として、『拾遺和歌集』のデータによって、プログラムの精度を高め、『小倉百人一首』で汎用性を確かめたと言ってよい。全130首のデータでは少なすぎるが、これは、今後より多くのデータを蓄積することにより、より精度を高め、汎用性を持たせるようにしたい。

まず、正答率であるが、前述のように、『拾遺和歌集』の場合、プログラムの精度を高めるために、そのデータを用いたものだから、当然正答率は高い。全80首、延語数1,146語で、誤答が27語、正答率97.6%という数値が得られた。また、『小倉百人一首』の場合、プログラムを全く手直しせずに、全50首、延語数737語、誤答が61語、正答率91.7%となった。この正答率は、本プログラムを組む前の目標（90〜95%以上）を一応超えてはいるが、まだ、不十分なものであることは確かである。

一方、処理時間については、非常に満足する結果が得られた。以前の BASIC 言語

では、配列変数内のデータの有無を調べるには、「FOR…NEXT」文を用いるしかなく、かなりの時間をこの箇所で費やすことがあったが、最近の BASIC 言語は、機能が強化、整備され、特に言語処理能力の面では、一段と飛躍した。よって、配列変数内のデータの有無を調べる命令語として、FBASIC-86 では、「SEARCH」命令を用いることにより、非常に短時間に判断を下してくれる。今回の作業では、文章データの単位区切りの始めから、1首中の全ての語句に対する品詞認定が終了したところまでを処理時間として計測した。その結果、『拾遺和歌集』『小倉百人一首』の合計130首、延語数1,883語を連続処理させたところ、延524秒を要した。ということは、1首あたりの処理時間は、約4秒、また、1秒あたりでは、約3.6語の語句を処理していることになる。この処理時間は、人間の能力をはるかに超えたものである。換言すれば、マイコンならではの速さと言えよう。ところで、この処理時間をより短縮する試みは、何度か行なった。しかし、この試みは、前述の正答率を高めるという試みと背反するもので、正答率を高めようとすると、それだけプログラムのステップが増加する。ということは、処理に時間がかかることになる。それに、BASIC 言語を使用する以上、今回得られた処理時間が限界と思われる。また、現在の私の方針としては、作業の目的が品詞認定にあるのだから、より正確な認定結果を得ることを優先しようと考えている。だから処理時間を多少犠牲にしても、この「品詞認定」プログラムの段階で正答率を高めておき、次の「品詞決定・漢字情報付加」プログラムにデータを引き渡すという方針を立てている。

6. おわりに

品詞認定は、人間にとってかなり高度な知的作業である。それをプログラム化して、マイコンにやらせるには、いくつかの壁があることがわかった。その最も大きなものは、同音異義語の処理である。例えば、「薄き衣ぞタチぞキてける」とある場合、「タチ」「キ」が「立来」なのか「裁着」なのかは、現在のプログラムのレベルでは判別できない。もし、こうした同音異義語を判別しようとすれば、意味的解釈や構文的解釈を取り入れるという方法を考えなければならない。しかし、コンピューターの性格からして、または、現在のコンピューターの性能からして、語句の意味的解釈を行なうことは不可能に近い。それよりは、構文的解釈の方がコンピューターには処理しやすいと言えよう。今回報告したプログラムにも若干取り入れはしたものの、この構文的解釈を取り入れるにも問題点がある。それは、事前に膨大な数の用例を収集、整理しなければならないという点と、それらの用例をディスク上に記憶させる場合、容量に限界があるので、ある程度の用例しか使えないという点等である。よって、当面の解決策がない今、この同音異義語の処理については、今後より適した方法を求めてゆきたいと思う。

その他に、現段階で残している問題として、今回作成したプログラム、特に「品詞認定」プログラムの汎用性の問題がある。今回行なった作業の初期段階から汎用性に

ついては、常に念頭に置いてきた。ただ、当面の必要性から、テストに使ったデータがすべて和歌であったため、散文について、どの程度の正答率が得られるものなのか、全く調査を行なっていない。予想では、和歌の場合とそれほど大差はないと思われるが、その点についても、今後綿密な検討を加えてゆきたい。

今回、このようなプログラムを作成しようと考えたのは、昭和46年3月刊行の国立国語研究所報告39『電子計算機による国語研究Ⅲ』所収の「品詞認定の自動化」（中野洋氏著）を目にしたのが契機である。中野氏は、HITAC-3010という大型コンピューターを使い、COBOL言語によってプログラムを組まれた。しかし、当時からすれば、今日のコンピューターの発達はめざましいものであるから、今から15年前の大型コンピューターが行なえたことなら、今の16ビットのマイコンでもやれるのではないかと考えたわけである。よって、品詞認定過程を辿る基本的な考え方やプログラム化するに際しては、中野氏の御高論およびフローチャートを参照させていただいた。また、プログラム内に取り入れた文法的事項は、基本的には中央図書刊『最新古典文法便覧』（遠藤嘉基氏・塚原鉄雄氏共著）に負うところが大である。ここに記して、深甚の謝意を表する。

◎

末尾ながら、本学名誉教授原田芳起、久保重両先生の傘寿、喜寿を、さらには本学教授木村三四吾先生の古稀を寿ぐとともに、3先生の今後の御健康と御活躍を祈念するものである。