マクロ方式かな漢字変換の実用性評価

マクロ方式かな漢字変換の実用性評価†

久野 靖††

計算機を使用して日本語文書を作成する方法の一つとして，かな分ち書きテキストから出発し，テキストごとに固有の辞書を用意し，つづり単位の置換えによって漢字かなまじりテキストを生成する，というものがある．このかな漢字変換方式をマクロ方式と呼ぶ．その特徴は，かなテキストを基本とするためテキストを能率よく入力，編集できること，および変換方式が単純なため処理速度が速いことである．しかるに，マクロ方式では利用者自身がテキストごとの辞書を管理し，同音異語に対する処置を行うので，その手間がどのくらいかは重要な問題である．そこで本文では，翻訳書1冊分のデータを分析することにより，この問題に対する解答を与えることを試みる．その結果，利用者が管理しなければならない辞書の項目数は書籍1冊でも数千程度で，しかもある程度以上大きなテキストについてはその数はテキストの大きさのせいぜい平方根程度でしかふえないこと，およびテキストの全つづりのうち，同音異語に対する処置を個別に行わなければならないものの比率は約1.2%であることがわかる．マクロ方式は用語の統一やつづり誤りの検出などを通じて文章を改良していく道具としても有効であるので，そのことを考え合わせると文章作成を中心とした日本語文書処理において有望であると結論される．

1. はじめに

近年計算機による日本語処理が一般化し，多くの人人がその恩恵に浴しはじめているが，現在使われている日本語処理システムはまだ検討の余地がある．実際，それらはほぼ例外なく

（1） 何らかの国字*入力方式により計算機に国字テキストを入力する，

（2） 国字テキストに計算機内で種々の処理を施す，

（3） 処理して得た国字テキストを出力する，

という手順をとっている．これは一見ごくあたりまえのようであるが，実はこれだけが唯一のやりかたではない．こうでないやりかたの一つとしてマクロ方式によるかな漢字変換[1]があげられる．マクロ方式の処理手順は

（1） 入力をかな分ち書きテスト（またはそのローマ字表現）により，ひとまずそのまま計算機に入力する（分ち書きされた各単語を以下つづりと呼ぶ），

（2） 計算機内部でも基本的にはかな表現のままで処理を行う，

（3） テキストごとに専用の辞書を用意し，出力時につづりの統一的置換えによって国字表記に変換する，

（4） 同音異語の区別は**確認語** (checkword)[2]と称する付加情報をテキスト中に含めることによって行う，

というものであり，扱う文章の種類によってはこのほうがまさっている点も多い．実際，われわれはこの方式を使って，これまでに翻訳書1冊[3]のほか，論文，学会の予稿にいたるさまざまな文書を作成してきたが，その経験からいってこの方式は大変使いやすいと感じている．

そこで本文では，このわれわれの感じを裏づけるため，訳書[3]の原稿作成に際して作られたかなテキストと辞書ファイルを分析することにより，マクロ方式によって文章を作成する場合どのくらいの手間がかかるかを，客観的に評価する．

マクロ方式の場合，テキストそのものの入力はたんなるかなタイプ（またはローマ字タイプ）であり，非職業的タイピストにも十分高速で入力が可能である．したがって，いちばん問題になるのは

（1） 辞書を作成，管理する手間，および

（2） テキスト中で同音異語の処置を行う手間

がそれぞれどのくらいか，ということである．以下ではまず2章においてマクロ方式の原理と特徴について説明し，3章ではマクロ方式において重要な意味をもつかなテキストの分ち書き方式について述べる．続いてかなテキストの分析に基づき，4章では利用者が管理しなければならない辞書項目数，また5章では同音異語の現れかたとその処理にかかる手間の分析を行う．最後に6章ではこれらの結果を踏まえてマクロ方式の総合的な評価を行う．

† Evaluation of Macro-Based Kana-Kanji Conversion by YASUSHI KUNO (Faculty of Science, Tokyo Institute of Technology).

†† 東京工業大学理学部

* 以下本文では日本語文用の文字を総称してこのように呼ぶ．

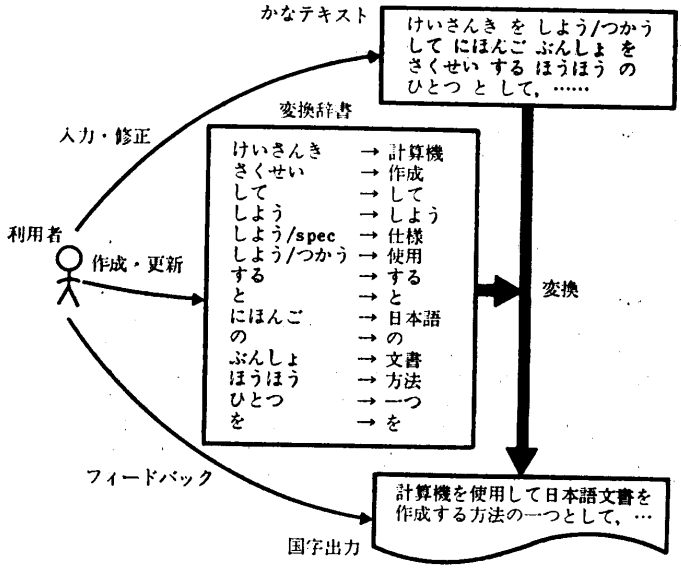

図 1 マクロ方式による国字テキスト作成
Fig. 1 Overview of the conversion process.

2. マクロ方式の概観

マクロ方式の原理については先に述べたとおりであるが，このあとの所論を理解しやすくするため，マクロ方式によれば本文の冒頭部分がどのように処理されるかを**図 1** に例示しておく．すなわち，利用者はかな分ち書きテキストとテキスト専用の変換辞書をもち，それらを種々の道具を利用しながら更新していく．同音異語をもつつづりに対しては確認語を使用する．確認語はつづりの直後に「/」で区切って書く．つまり，確認語つきのつづりは

つづり/確認語

の形をとる．確認語は利用者が自然でおぼえやすいと思うものを自由に選んでよい．確認語はかなテキストとテキスト専用辞書の両方で用いられる．たとえば，「使用」，「仕様」は，かなテキスト中では「しよう/つかう」，「しよう/spec」のように確認語をつけて書き，変換定義も「しよう/つかう→使用」のように確認語を含めて定義する．

本方式では，「書かない」，「書けば」のように一つの用語「書く」から派生したつづりも，つづりとしてはまったく独立に扱われる．このため，訳書[3]の作成の際は一つの用言のさまざまな言いまわしごとに別の変換定義を作成しなければならず，その手間はかなり大きかった．のちに語幹と活用語尾を含む辞書を参照して活用形の変換定義を作り出す機能を追加した[4),5)]ため，現在ではこの問題点は解消されている．ここで注意すべきことは，現在でも変換辞書自体はつづりそのものとその国字表記の対応を示す形を保っていることである．そのため，変換処理が高速で実施でき，利用者からみても単純でわかりやすく，変換誤りに対しても確認語を用いれば容易に対処することができる．

実際に本方式を用いてある程度大きな文書を作成した例として，訳書[3]の翻訳書作成がどのような手順で行われたかを簡単に説明する．作業が処理系の開発と並行して行われたため，結果的にまわりみちとなったことも多かったが，ここでは細部は省き，大筋のみを述べる．

（1） テキストの打込み．訳書[3]の作成においては，かなテキストは訳者の手書き原稿を多人数で分担して打ち込むことにより作成された．いったん手書き原稿を用意したのは，当初はシステムの実用性について見通しが得られていなかったため，および訳者の手もとに端末がなかったためであり，現在ではむしろ訳者が直接訳文を考えながら入力するほうが楽であると思われる．

（2） 分ち書きの統一．（1）で作成したテキストは多人数で分担して作ったこと，および分ち書きについて意見が不統一であったことのため，そのままでは効率よく変換できないことが明らかであった．そこでテキスト中に現れるつづりを統一的に置き換える道具を作成し，この助けを借りて分ちを統一する作業を行った．これについてはあとで詳しく述べる．

（3） 辞書作成．テキスト専用辞書は，以前にテキストの一部について試験的変換のために作成したもの[10),11)]を利用し，不足項目を追加する，という形で行った．

（4） 反復推敲．この段階までで作られたテキストと専用辞書を用いて変換を行い，結果を漢字プリンタで打ち出して推敲した．この段階では変換誤りの除去だけではなく，文章自体の修正，改良も相当量なされた．

（5） 最終チェック．通常は漢字まじりの出力が満足のいく状態になれば校了となるのであるが，訳書[3]ではさらに念を入れて，同音異語をもつつづりにつき，KWIC 索引を打ち出してチェックを行った．この結果，変換誤りとまではいえないようなこまかい点まで厳密な表記の統一を行うことができた．

通常の文書作成ではここまで厳密な手順を踏む必要はないと思われる．とくに（2）については，現時点か

らふりかえって見ればマクロ方式について一応理解した人が打ち込めばほとんど分ちの調整は不要と考えられる（後述）し，（5）のようなこともとくに高品質をねらわない限り必要ない．また，（3）の辞書作成も現在では自動作成機能が使えるため，ゼロから出発する必要はなくなっている．したがって現在の利用形態では次のような手順が標準的である．

（1） 文書の構想がある程度まとまったところで端末の前へ行き，テキストを一通り打ち込む．

（2） 辞書を自動作成する（必要ならこの段階で辞書をチェックして，余分な定義を消したり不足するものを追加したりできるが，とくに何もしなくともかまわない）．

（3） 変換を行い，出力を見ながらテキストおよび専用辞書の追加，修正を行う．満足のいく結果が得られるまでこれを繰り返す．

3. かなテキストの分ち書き方式

このようにわれわれは，かな分ち書き日本語文から出発するが，日本語文における分ち書きの仕方は英文の場合のように明確には定まっていない．一方，マクロ方式かな漢字変換では，どのように分ち書きを行うかが，原理的に重要な意味をもつ．したがって分ち書きの方式については何らかの原則をもうけることが必要である．

マクロ方式では利用者はたえずかなテキストに接していることになるので，自然で読みやすい分ち書き方式を選ぶことが必要である．また，マクロ方式では各つづりごとに変換定義を用意するので，つづりの種類数が少ないことが望ましい．さらに，利用者にわかりにくい規則を押しつけることは好ましくないので，規則をできるだけ単純化する必要がある．そこでわれわれは

（1） 各つづりは「単語」であるとする．単語とは国語辞典に見出し語として載っているようなものをいう，

（2） ただし，切り分けて書くと不自然なものに限っては，つなげて書く，

という原則を採用することにした．この原則にしたがって助詞は名詞とは切り離して書き，動詞の活用語尾などは単語の一部なので続けて書く．助動詞は本来はそれがついている用言とは別のものであるが，切り離して書くと不自然なので，つなげて書く．問題は接頭語，接尾語，数詞などを含む言葉，あるいは複合名詞のように全体としてひとまとまりでありながら内部に構造をもつものである．結局このような場合には利用者が自然と思う方に統一してあればよいことにした．

なお，前述のように，訳書[3]の作成において最初に打ち込まれたテキストの分ち書きの仕方はかなり不統一なものであった．そこでテキスト中のつづりを広域的に置き換える道具を用いて分ちの不統一を直す作業を行った[6]．その際見つかった長すぎるつづりの例としては「われわれはそのような」，「まださわってないとき」などがあり，逆に切りすぎの例としては「よみこん」（よみこんだ，よみこんで），「うごい」（うごいた，うごいて），などがあった．

この作業に要した時間は合計約19時間であったが，その結果テキストに含まれるつづりの数とその種類数は次のように変化した．

	つづり合計数	つづり種類数
修正前	121,490個	10,960種
中間結果	149,506	7,673
最終結果	147,702	7,769

このように，分ちが不統一だとテキストに含まれるつづりの数が不必要に大きくなる．実際，テキストの中に出てくるつづりは本質的には8,000種ほどであったのに，もとの状態では分ちのゆれのために3,000もの余分なつづりが派生していたことになる．もちろん，これらすべてが漢字を含む変換定義と関連しているわけではないので，そのまま漢字への誤変換にはつながらない．しかし，辞書の管理の手間を考えれば分ちの統一が不可欠なことは明らかである．また分ちを統一することにより，かなテキスト自体もずっと読みやすいものとなった．

つづりの種類数の減少が，分ちかたを不当にこまかくすることによって得られたものではないか，という疑問もないではなかったが，（表からもわかるように）まず最初にできるだけ細かく切りわけ（中間結果），次に切りすぎのものをつなぎ合わせるという方針で進んだところ，2,000箇所近くつなぎ合わせをしたにもかかわらず，つなげる前と後ではほとんど種類数が変わらなかった（差は100個弱）ことから見て，分ちかたがある程度統一されれば，それ以上切り分けてみても種類数はあまり変化しないものと思われる．すなわち，上記の種類数の減少は大部分分ちかたの統一によって得られたといえよう．

一方，訳書[3]の完成以降は，はじめてこの方式を使う利用者に対しては，

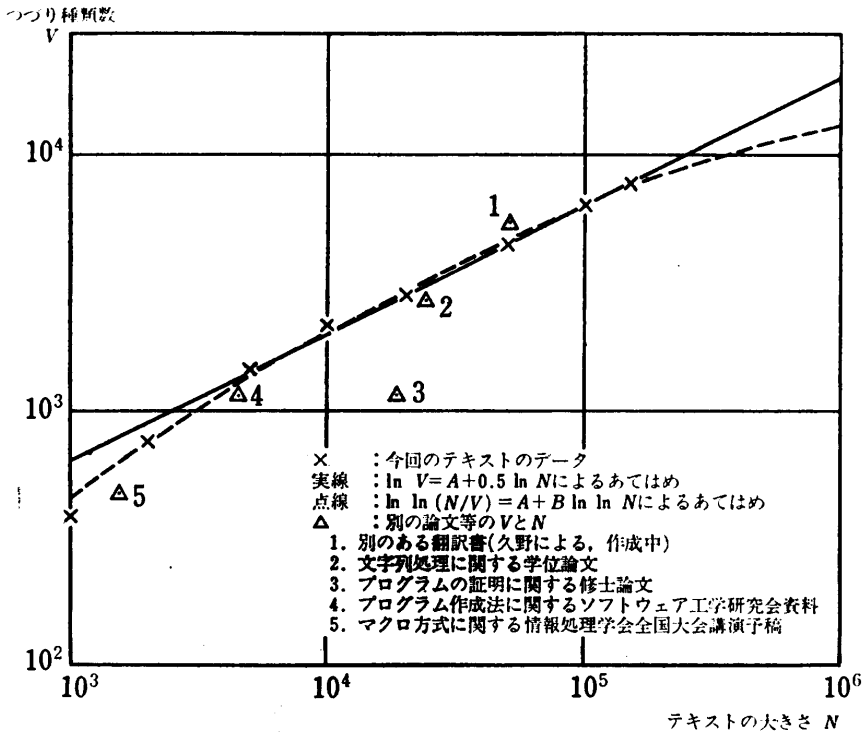

図 2 テキストの大きさとつづり種類数の関係
Fig. 2 The relation between text size and vocabulary.

（1） マクロ方式の原理について説明する
（2） 上記の分ち書き規則を説明する
（3） テキストの見本を見せて具体的にどうやっているかを教える

といった程度の内容のガイダンス（5～15分くらい）を行ってきたが，これらの利用者が作成したかなテキストにおいても，分ち書きの不統一による問題はほとんどなかった．試みに，英文タイプはできるが本方式の経験はない女性秘書に上記のガイダンスを行ったあと，3,800字程度の文章（内容は計算機に関係のある読み物）をローマ字で打ち込んでもらって，分ち書きの誤りを数えたところ，結果は次のようになった．

つづり数	1,776語
分ち誤り	32箇所
（うち，変換に関係）	10箇所

すなわち，分ち誤りの数はつづり数の1.8%とかなり少ない．しかも，そのすべてが漢字への変換に関係してくるわけではなく，誤変換につながるものに話を限れば10箇所，全つづり数の0.56%程度にすぎなかった．このことから見ても，分ち誤りの頻度は初心者においても十分低く，それによるオーバヘッドはほとんど無視できる．

また，マクロ方式では必ずしも「100%統一的な」分ち書きを利用者に要求するわけではないという点も注意を要する．たとえ分ちに「揺らぎ」があっても，対応する定義があれば問題なく変換できる．実際，訳書[3]のかなテキストにおいてもこの種類の「揺らぎ」はいくつか残っている．

4. かなテキストにおけるつづりの性質

訳書[3]のかなテキストの大きさは最終稿で151,180つづり，つづりの種類数は7,781になったが，マクロ方式ではつづりの種類数が大きくなるほど辞書を管理する手間がふえるので，一般の文書についてテキストの大きさとつづりの種類数がどのような関係にあるかが重要な問題となる．

図2 ×印は訳書[3]のかなテキストについてテキストの最初から数えた通算語数（大きさ）とそこまでに現れたつづり種類数を両対数目盛りでプロットしたものである．これを見ると，テキストの大きさが5,000つづりくらいから先ではグラフの傾きが0.5くらいになっている(実線)．つまりつづり種類数はテキストがある程度大きくなったところではテキストの大きさの平方根程度でしかふえていないことになる．さらに，より精密なモデルとして文献8)のものを使ってあてはめを行ったものを点線に示してある．このモデルはテキ

ストの大きさを N, つづりの種類数を V としたときこれらの間に

$$\ln\ln(N/V)=A+B\ln\ln N$$

の関係があることを仮定したもので，複数の言語についてかなり大きなテキスト（$N=200,000$ 程度）までよくあてはまり，外挿にも適したものとされている．このモデルを信じれば，テキストの大きさがより大きいところではつづり種類数はテキストの大きさの平方根よりもさらに少ないオーダでしかふえない．たとえば訳書[3]を書きたして現在の大きさ（和文300ページ）の2倍にしたとしても，つづりの種類数は文献8)のモデルでは9,300，0.5乗でふえていったとしてもたかだか11,000程度ですむ．

以上は訳書[3]という特定のかなテキストに関する話であるが，同様のことが一般のテキストについてもいえるかどうかを調べるために，マクロ方式によって作成した五つの別個の文章についてそのテキストの大きさ（総語数）とつづり種類数を図2△印で示す．これを見れば訳書[3]のテキストに基づく曲線は一般の技術文書についてもおおむねあてはまるものと考えてよさそうである（文書3だけはかなり大きくはずれているが，これは文書がプログラムの証明に関する論文でそのかなりの部分が論理式等で占められているという特殊性によったものと思われる）．したがって，技術文書に関していえば実用上十分な範囲の大きさの文書について，テキストの大きさがある程度以上大きければつづりの種類数はたかだかテキストの大きさの平方根のオーダ以下でしかふえない，と推測される．なお，文章1～5を訳書[3]のかなテキストのうしろにつけ足したものを考えた場合，その全体としてのつづり種類数は0.5乗のモデルに従って外挿した値よりも，つけ足した文章のつづり種類数の10～20%程度上になる．これから，つづりの種類数がテキストの大きさがふえてもさほどふえないのはテキストが一つの文書として内容的に一貫していることが大きな理由であると推測される．

次に，訳書[3]のかなテキストに最終的に現れた約8,000のつづりの内訳を示すと次のようになる（前述のように，同じことばから派生したものでも，つづりとして異なるものは区別している）．

英単語	1,874種	(23.9%)
カタカナ	197	(2.5%)
漢字まじり	4,246	(54.7%)
ひらがなのみ	1,467	(18.8%)

英単語が多いのは，内容の大半がプログラムとその解説であるという特定の書物の性質による．英字やカタカナのみのつづりは漢字まじりに変換されることは（常識的な使いかたでは）ありえないので，利用者が「面倒をみる」必要がある辞書の項目数は約5,700ということになる．

辞書に漢字まじりに変換されるもののみを入れておくことにすれば，この数はさらに20%ほど減らせるが，すべての「正しい」つづりを辞書に入れておくことによって「これまで存在しなかったつづり」を自動的に見つけ出し，つづり誤り検出や辞書項目の自動的な追加などに役立てることができるので，むしろ積極的に「ひらがなである」という情報を利用したほうが有利である．この考えを最初に導入したのは杉村[10]であるが，当時から比べると，分ちの統一により，辞書に載っていないものの比率がいっそう少なくなったのでこの考えかたの有効性はさらに増したといえる．

もっとも，たとえ20%でもひらがなとわかり切っている付属語などが多ければ，利用者はそのような「ごみ」を大量に含む専用辞書を見なければならないことを苦痛に感じるかもしれない．しかし実際には，このひらがなのみのつづりのうち，付属語（助詞，助動詞）は63種類にすぎなかった．

さて，一般の文書について考えた場合，新しく文章を書くたびに変換定義を白紙の状態から作成するのでは大変である．そこで，あらかじめ「標準的な」定義の集り（国語辞典のようなもの）を用意しておくことが考えられる．その有効性を調べる目安として，先につづり種類数について調べた文書のうち，文書2～5においてその変換定義の中で訳書[3]のかなテキストのために作成した変換定義と共通する部分がどのくらいあるかを，漢字を含む表記に対応したものとそれ以外

表1　四つの文書の変換定義と訳書[3]のかなテキスト用変換定義との共通部分の比率
Table 1　Number of common definitions for other sample text.

文書	つづり合計数	つづり種類数	共通部分 (非漢字)	共通部分 (漢字)	それ以外 (非漢字)	それ以外 (漢字)
2	24,230個	2,768種	564種 (20.4%)	989種 (35.7%)	533種 (19.3%)	682種 (24.6%)
3	19,194	1,261	248 (19.7%)	375 (29.7%)	302 (23.9%)	336 (26.7%)
4	4,489	1,127	328 (29.1%)	526 (46.7%)	140 (12.4%)	133 (11.8%)
5	1,544	467	163 (34.9%)	204 (43.7%)	44 (9.4%)	56 (12.0%)

の2群に分けて集計したものが**表1**である（なお，これらの文書の著作者はいずれも訳書[3]の訳者とは異なる）．すなわち，これらの文書は「計算機に関係がある」という以外にはほとんど共通する話題をもたないにもかかわらず，変換定義の約半数以上は訳書[3]のものと共通しており，新たに追加しなければならない漢字まじりの定義（「それ以外」の「漢字」）はさほど多くない（つづり種類数の30%以内）．このことから「標準的な」辞書を整備することでかなり新規に作成する定義の数を減らせることがわかる．用言処理を組み込み，大規模な標準辞書を用いた場合については現在研究中であり，別の機会に述べる．

5. 同音異語とその処置

前に述べたように，マクロ方式では同音異語がある場合にはつづりに確認語を付加することであいまいさを解消している．どのような規則に従って確認語をつけるのがよいかどうかは初めのうち必ずしも明らかでなかったが，翻訳書の作成作業を通じて得た経験に照らして現在では次のような方式をとることにしている．

（1） 同音異語がある場合は原則として漢字まじり表記に対応するものはすべて確認語をつけ，ひらがな表記に対応するものにはつけない．

（2） ただし，同音異語のうち一つだけが圧倒的に多く現れるなら，それを確認語なしのつづりに対応させてもよい．

見てわかるように，これらの規則は決して「確認語の数を最小化する」ものではなく，むしろ利用者から見て自然で不愉快な思いをすることが少ないことをねらったものである．

たとえば訳書[3]において，同音異語をもつつづりのうち，各同音異語が同じ程度現れた例として，次のようなものがあった．

読んで	24回	呼んで	15		
種	56	主	44		
書き	23	下記	10		
仮定	30	過程	17		
しよう	56	使用	38	仕様	15

これらの場合，たとえば「読んで」が「呼んで」よりいくらか多く現れるからといって「よんで」を「読んで」に対応させてしまうと本来「呼んで」であるべきところが「読んで」になってしまうという誤りがたくさん起こることになる．これは利用者にとって大変不愉快である．むしろ「よんで」はかなのままで出力され，その結果を見てたとえば「よんで/よむ」や「よんで/よぶ」のように適当な確認語をつける，というほうが自然であろう．

一方，同音異語の中には，上のようにそれぞれの場合が同じくらい現れるものばかりではなく，どれか一つが圧倒的に多いものもかなり見受けられた．その例としては

以上	102回	異常	4
以外	107	意外	2
あった	101	合った	1

などがあげられる．これらでも，いちばん多いのがひらがな表記である場合は(1)の原則をそのまま適用すればよい．また，漢字同士の場合でも，まずまとめて多いほうに変換させ，間違ったところだけ修正することができる．したがってこのような場合は利用者が個別に処置をするのは「少数派」の場合だけといえる．

次に訳書[3]のかなテキストにおいて上記の原則に従ってつけた確認語の内訳を示そう．訳書[3]のテキストでは，種類にして約170のつづりが同音異語に関係していた．これらのつづりは実は350種類の言葉に対応しているが，かな表記の上では互いに重なって170種類になっているものである．この特定のかなテキストに関する限り，ほとんどの場合一つのつづりはたかだか2種類のことばに対応しており，3種類以上のことばに対応している例は**図3**に示す計17組にすぎない．

一方，上記350種類のことばの出現数の合計は約16,000回であり，したがって150,000個のつづりのうち10%以上が同音異語に関係していることになる．もしこれらすべてに人間の手で確認語をつける必要があるとしたらとうていこのシステムは使いものにならないであろう．しかし，前述のように，利用者は必ずしも上記16,000個について一つ一つ処置をする必要はない．同音異語の組は処置の手間から見て次のように分類することができる．

（a） 同音異語のうちただ一つの使われかたが9割以上を占めているもの—先に述べたように，個別の処置は少数の「例外的な」使われかたの数だけ行えばよい．

（b） ひらがな定義を含むもの—確認語なしのものに対してはとくに手を加える必要がなく，確認語をもつものについてだけ個別に処置をすればよい．

（c） それ以外のもの—すべてのつづりに対して適切な確認語を判断して付加する必要がある．

おき	→ おき	げん	→ 原	しき	→ しき
おき/place	→ 置き	げん/げんざい	→ 現	しき/いしへん	→ 磁気
おき/おきる	→ 起き	げん/みなもと	→ 源	しき/とき	→ 時期
か	→ か	こう	→ こう	たいして	→ たいして
か/ばける	→ 化	こう/term	→ 項	たいして/たいする	→ 対して
か/べし	→ 可	こう/たかい	→ 高	たいして/だい	→ 大して
かい/あらため	→ 改	さい	→ 際	とく/うる	→ 得
かい/した	→ 下位	さい/ちがい	→ 差異	とく/かい	→ 解く
かい/まわる	→ 回	さい/ふたたび	→ 再	とく/ごんべん	→ 説く
かえって	→ かえって	し	→ し	は	→ は
かえって/return	→ 帰って	し/うじ	→ 氏	は/さんずい	→ 派
かえって/しんにゅう	→ 返って	し/さっし	→ 誌	は/はっぱ	→ 葉
				は/やいば	→ 刃
かん/あいだ	→ 間	しよう	→ しよう		
かん/こころ	→ 感	しよう/spec	→ 仕様	やく/だいたい	→ 約
かん/まき	→ 巻	しよう/つかう	→ 使用	やく/ほんやく	→ 訳
				やく/やくわり	→ 役
き	→ き	し/あざ	→ 字		
き/け	→ 気	し/つぎ	→ 次		
き/はた	→ 機	し/つち	→ 地		
き/もく	→ 木	し/とき	→ 時		

図 3　三つ以上のことばが重なった同音異語
Fig. 3　Duplication of more than three words.

訳書[3]のかなテキストにおける170組の同音異語をこの基準に従って分類したものを次に示す.

	種類	統一的処理	例外
(a)	50種	12,800回	280回
(b)	65	1,960	630
(c)	55	—	860
合計	170	14,760	1,770

すなわち，種類に関しては(a)～(c)はほぼ同じくらいの数であるが，つづりの出現数について見ると，(a)のうちの統一的に処理できるものの数が圧倒的に多数を占めている．これに(b)でとくに手を入れなくてよいものの数を加えるとその合計は15,000近くになり，人間が個々に面倒を見なければならないのは1,800箇所程度，割合にして全つづりの1.2%くらいですむことになる．

なお，最終的に確認語を必要とする1.2%についても，テキストの打込みと同時に確認語をつける必要はなく，ある程度テキストがたまってきてそろそろきれいに打ち出したいと思ったとき，初めて問題のあるつづりのみに注目して確認語をつければ十分である．

実際，訳書[3]のかなテキストの場合も，最初は確認語はまったくついていなかった．しかし，出現つづりと漢字表記の対応表（以下単語帳と呼ぶ）を打ち出して眺めたり，部分的に変換をしてみるうちに，どのつづりに同音異語が存在しそうかはだいたい見当がつくようになった．そこで3章で述べた，つづりを統一的に置き換えて直す道具を利用して，同音異語のありそうなつづりにまとめて「/?」という目印をつけ，続いてテキストの先頭からエディタでこの目印をさがしてはそれぞれ「?」を適切な確認語で置き換える，という形で作業をすすめたところ，ごく容易に適切な確認語をつけることができた．したがって実際に確認語をつける手間は想像するよりずっと少なくてすむ．さらに，同音異語のあるつづりについては最後にまとめてその付近を打ち出してチェックしたため，結果的には手で原稿を作成する場合よりも誤りがずっと少なかった．このことはエディタを含めた道具の重要性と，それらを活用した場合の文章の質を改良する道具としてのマクロ方式の可能性を示しているといえる．

このように，1.2%という数字は十分少ないものであると考えられるが，テキストの文脈情報を利用することによって，いっそう，確認語の必要性を減らすことも考えられる．たとえば「使用」と「仕様」の場合，「使用する」とはいうが「仕様する」とはいわないので，次にサ変動詞がくるかどうかである程度これらの場合を見分けることができる．そのほか形容動詞，格助詞，接頭語，接尾語などのさまざまな情報を取り入れることで処置の必要な場合を最大全つづりの0.5%くらいまで減らせるという見積りを得ている[6]．しかし，そのようなことをすれば当然処理時間は多くかかるし，処理方式が複雑になるため，利用者から見て処理系の働きが透明でなくなる危険が大きい．これらの欠点を承知の上で文脈に関する処置を行うべきかどうかは今後研究の必要がある．

6. マクロ方式の評価と今後の課題

これまでに述べてきたように，マクロ方式による日本語処理の特徴としては

（1） 実用的な速度で処理が可能である*，

（2） 変換原理が単純でわかりやすい，

（3） 入力，修正をかなテキストの形で行うので直接的で扱いやすい，

（4） 用語の統一やつづり誤り検出などにかなテキスト固有の情報が利用できる，

があげられる．（1）のおかげで実用的な道具として気軽に使うことができ，（2）のおかげで利用者が心理的安定感をもつことができる．また，（3）については，端末の前で漢字への変換のことは気にせずに文章を打ち込んでいけるのが最大の利点である．実際筆者らの経験によれば，簡単なメモ程度をもって端末の前にすわることで，ごく短時間にかなりな量の文章を創作して打ち込んでしまうことが可能である．そして，そのようにして作った文章を組み換えたり改良したりすることも，ふつうのテキストエディタで十分能率よくできるので，無理に最初から完全な文章を作ろうとしないですむ．

かなテキストというと一見国字テキストより少ない情報しかもっていないように思えるが，そうとばかりはいえない．マクロ方式のかな分ち書きテキストにはべた書きの国字テキストには含まれないさまざまな情報が含まれており，それらを容易に引き出すことができる．たとえばつづり誤りの検出については前にも触れたが，これはテキストがつづりに分かれているからこそ可能なわけであり，国字テキストで同じことをしようとすれば複雑な解析が必要になろう．またつづりはそれぞれことばに対応しているので単語張や索引の自動作成も容易である．単語帳が文章を改良する上で強力な道具になることはすでに述べたとおりである．

一方マクロ方式を用いる場合の問題点は

（1） 利用者が辞書を管理しなければならないこと，

（2） 同音異語があるとき確認語を用いる必要があること，

であるが，（1）については，テキストが大きくなっても辞書の大きさはさほどふえないため，やりかたをおぼえてしまえばさして問題にはならない．（2）も，実際にテキストの中で確認語をつけなければならないつづりは全つづりの1.2%ほどにすぎない．そして，マクロ方式ではテキスト作成と変換のための処置はそれぞれわけて行うため，問題のあるつづりをあらかじめ調べておいてまとめて処置すればよく，実際の手間は1.2%という数字から想像されるよりはずっと少ない．

もっとも辞書や確認語の管理が負担になるかどうかは，そのための道具がどれくらい整備されているかにもよる．筆者らの経験でも初期，道具が整備されていなかったうちは非常に労力が大きかった．現在では道具もそろって，システムとして形が整ってきているので，利用者の負担はごく小さい（用意した道具その他についての詳細については文献9）にゆずる）．これらから総合的に判断すると，マクロ方式による日本語文書処理は非常に有望であるといえる．今後残された課題としては

（1） より多くのデータを集め，経験を積むことによって，いっそうの裏づけを行うこと，

（2） 文脈情報を利用した同音異語処理について検討すること，

（3） 用言の語尾変化，および複合語の処理を行うことにより，辞書作成の手間を減らすこと（一部実現ずみ），

（4） システムとしてさらに使いやすくなるように道具や処理方式の細部を改良していくこと，

があげられる．

謝辞 本研究を行うにあたってご指導くださった木村泉教授に感謝します．

* FACOM 230-45S 計算機上の CLU 語で書かれた処理系で，訳書[3]のテキストを変換するのに要する経過時間は30分弱であった．

参考文献

1) 木村：マクロ方式のカナ漢字変換による訳書作成の経験と将来構想，第22回情報処理学会全国大会講演論文集，p. 869 (1981).
2) Gilb, T. and Weinberg, G. M.: *Humanized Input*, Winthrop Publishers, Inc., Cambridge (1977).
3) カーニハン，プローガー原著（木村泉訳）：ソフトウェア作法，共立出版，東京 (1981).
4) 久野，遠城：マクロ方式かな漢字変換のシステム化，第24回情報処理学会全国大会講演論文集，p.997 (1982).
5) 遠城，久野：マクロ方式かな漢字変換システムにおける標準辞書検索方式，第24回情報処理学会全国大会講演論文集，p. 999 (1982).
6) 久野，木村：マクロ方式によるカナ漢字変換の評価，第22回情報処理学会全国大会講演論文集，p. 871 (1981).
7) 遠城，久野，木村：マクロ方式のカナ漢字変換

における用言処理，第22回情報処理学会全国大会講演論文集，p.875 (1981).
8) Trudava, J.: A Mathematical Model of the Vocabulary-Text Relation, Proc. of the 8th International Conf. on Computational Linguistics, pp. 600-604 (1980).
9) Kimura, I., Kuno, Y. and Enjo, H.: A System for Japanese Authors Based on Bulk Phonetic-to-Ideographic Conversion, Proc. of 1983 International Conf. on Text Processing, pp. 403-408 (1983).
10) 杉村：東京工業大学修士論文 (1977).
11) 関根：東京工業大学修士論文 (1977).

(昭和57年 9 月20日受付)
(昭和58年12月13日採録)