# 連文節かん字漢字変換

4C−1

建石由佳・金子宏・鳥原信一

日本アイ・ビー・エム(株)東京基礎研究所

## 「かん字漢字変換」とは

「かん字漢字変換」は、日本文の表記変換の一種で、ユーザーの書きたい文Ｓを構成する漢字の一部がその読みに対応するかなになっている入力文からＳを生成する操作であるとする(図１)。Ｓを構成する漢字のすべてをかなにした文からＳを生成する操作がかな漢字変換である。かん字漢字変換は、入力に漢字が交じることを許した、かな漢字変換の拡張であると見ることもできる。

かん字に変かん　→　漢字に変換

(図１)かん字漢字変換

かん字漢字変換は、直接漢字が入力できるフロントエンドの補助手段として有効である。実際、タッチタイプ入力の補助手段としてかん字漢字変換を用いることが研究・実現されている([1], [2])。最近、ペンを用いて日本文を手書き入力するシステムが市場に現れている。市販システムでは、ユーザーが忘れてしまった漢字やうまく書けない(認識されにくい)漢字を含む単語を、かな漢字変換により入力する機能が設けられているが、このようなシステムでも、かん字漢字変換を用いて、書ける漢字はそのまま書くことを許せば、かな漢字変換を用いる場合に比べて入力を速く、快適にすることが期待できる。

## かん字漢字変換の実現方法

われわれは、連文節かな漢字変換の辞書を、漢字を含んだキーで検索できるように拡張して、かん字漢字変換を実現した([3])アルゴリズムは、連文節かな漢字変換のアルゴリズムをほぼそのまま用いる。

## かん字漢字変換の出力選択

日本語には同音異義語が存在するので、ある入力に対するかな漢字変換の出力は一意ではない。コスト最小法かな漢字変換では、一般に、出力候補のそれぞれにコスト付けをして、その最小となるものを出力としていた。かん字漢字変換でも、一般にある入力に対する出力は一意ではないので、なんらかのコスト付けをして、候補を選ばねばならない。かん字漢字変換では、かな漢字変換と異なったコスト付けが必要となる。また、その変更は、単にコストの値を調整すれば良いというものではなく、コスト付けの方法そのものを変えなければならない。

かな漢字変換では、一般に、出力候補のみに依存して、それぞれにコスト付けをすることができた。それは、一つの出力に対する入力、すなわち、その出力をかな書きしたものが、一部の例外を除いて一意に定まったからである。しかし、かん字漢字変換では、一つの出力に対して入力表記がいくとおりも考えられる。このとき、同音異義語の出力の選択の中で、入力表記によってコストを変える必要が生じる場合が存在する(図２)。

| | 入力 | 出力 |
|---|---|---|
| 1. | ごじゅう | 五十 |
| | 五じゅう | 五重 |
| 2. | ふくしゅう | 復習 |
| | 復しゅう | 復讐 |

(図２)表記によって優先順位が変わりうる

例１．では、一般には「五重」より「五十」のほうが高頻度であるので、かな漢字変換においては「五十」を優先させるコスト付けにするのがよい。しかし、「五じゅう」という表記では、むしろ「五重」を優先させるのが自然であろう。数字「五十」を書くのにわざわざ単位ごとに表記を違えて書くことは少ないと思われるからである。例２でも、一般には「復習」のほうが高頻度であるが、手書き入力で、文字が書きにくいという理由でかなで入力する可能性は、「習」よりもむしろ「讐」のほうが高い。このように、入力の表記法によって、出力の優先順位が変わりうるので、出力だけではコストを決められず、出力と入力表記の両者に依存して決めねばならないことがわかる。

また、かな漢字変換では対立しなかった出力どうしが、一部を漢字書きすることによって読みの情報が失われるため対立候補となるようになる。これは、「同表記異義語」と呼べるものである(図３)。

| | | | |
|---|---|---|---|
| せい子 | → | 精子 | |
| | → | 聖子 | ([2]による) |

(図３)同表記異議語

さらに、文節単位、連文節単位で変換を行う場合、次のような組も「同表記異議」となる。

---

KANji-to-Kanji Conversion: Kana-to-Kanji Conversion with Partially Converted Input

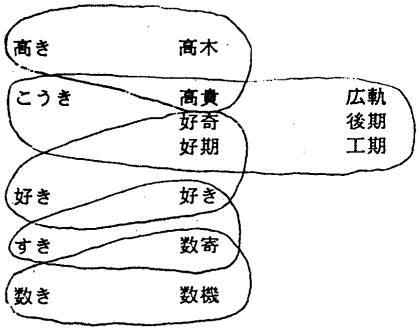
Yuka Tateishi, Hiroshi Kaneko, Shin-ichi Torihara
IBM Research, Tokyo Research Laboratory

| | | | |
|---|---|---|---|
| 苦しんで | → | 苦しんで | |
| | → | 苦心で | … A |
| わたしだけが家でした | → | わたしだけが家でした | |
| | → | わたしだけ画家でした | … B |
| | → | わたしだけが家出した | |
| 長い間 | → | 長い間 | |
| | → | 長居間 | … C |

（図４）連文節変換における同表記異義

A は活用語尾ひらがな書き部分が、同じ漢字で別の読みを含む語の一部と見なされた例である。B は、単文節変換では問題とならないが、左側に助詞「が」があるために、A と同様、「家」の別の読みを持つ語の一部となったものである。C では、「長」の読みは左右で変わらない。この場合、「間（あいだ）」が漢字になったため、「間（かん）」という読みの可能性ができ、その結果「ながい（形容詞）あいだ（名詞）」と「ながい（固有名詞）かん（接尾）」が対立するようになったものである。

同表記異議語の問題点は、対立候補の集合が増えることである かな漢字変換の場合、ある出力 S に対する対立候補は、S のみで決まる。しかし、かん字漢字変換では、S の表記として許されるかなと漢字の組み合わせの数だけ入力がありうる（図５）。ここで入力表記に依存せずに出力だけでコストを決めようとすると、かえって多くの出力候補がコスト決定に関わることになる。「好き」のコストの値を決めるのに、かな漢字変換では同音の「数寄」との対比だけで良かったが、かん字漢字変換では、同表記意義となる「好奇」のコストも考慮にいれる必要が生じる。さらに、「好奇」のコストを決めるために「高貴」「好機」「広軌」など、「数寄」のコストを決めるのに「数機」など、といった広い範囲の依存関係ができるからである。




（図５）かん字漢字変換における対立候補

ここで問題となるのは、コストの大小関係がループになっている組の存在が否定できないことである（具体例はみつかっていない）。ループになっている組、とは図６のような図式で表される組である。図２の例は、ユーザーが入力によって出力の選択を変えようという意図を持っていることを仮定していた。しかし、このような組では、ユーザーがそのような意図を全く持たなくても、出力どうしの優先順位が決められなくなる。

| 入力 | 出力 | |
|---|---|---|
| A | X | |
| B | X | Cost(X)>Cost(Y) |
| B | Y | |
| C | Y | Cost(Y)>Cost(Z) |
| C | Z | |
| A | Z | Cost(Z)>Cost(X) |

（図６）コストの大小がループになる？

X，Y，Zが同音であれば、それらのコストは同音異義語選択のさいのコストと同じであるから、ユーザーの意図を除けば、出力のみでコスト付けができる。したがって、このようなループが問題になるのは、同音でないものが同表記になるときである。

このような組で、コストの大小関係がループになるのは、同じ出力に対するコストは入力にかかわらず同じ値としたことによる。このような現象が起こらないといえないことから、ユーザーの意図による選択、という要素をぬきにしても、かん字漢字変換のコストは、出力だけではなく、入入力表記と出力の両者に依存して決めねばならない。

以上のように、かん字漢字変換では、出力と入力表記の両者に依存したコスト付けが必要になるが、このとき、入力表記と出力の組を辞書中の一単位として、それぞれにコストを割りふる方法では、ユーザーの意図による対立に対してうまく働かない。なぜなら、入力表記に関わる要素は、ユーザーのくせ（たとえばどんな字を覚えているか）、文字認識プログラム（外字、認識率の悪い文字など）など、かん字漢字変換の外部に大きく依存するからである。

また、かな漢字変換の同音異義語選択の場合と同じく、学習によって優先順を制御することも考えられる。この場合、学習の単位をどうするかが問題となる。出力だけを学習することにすると、上の問題は解決されない。一方、入力と出力の組を学習することにすると、新しい（辞書にない）語に対して、特定の表記でしか学習できないことになる。

## まとめ

かな漢字変換の入力に漢字が交じることを許した「かん字漢字変換」を作成した。かん字漢字変換では、入力表記に自由度があるので、入力の表記もコスト関数の引数として扱わなければならず、かな漢字変換に比べて候補選択問題が複雑になる。今後、応用面での制約を考慮しつつ、一般的に有効な候補選択方法について研究していきたい。

## 参考文献

[1]喜多ほか:「２ストローク入力用仮名漢字変換システム，情報処理学会・文書処理とヒューマンインターフェース研究会 16-4, 1988.1
[2] 小野:「Ｔコードの補助入力:字形組み合わせ法と交ぜ書き変換法」，情報処理学会論文誌 Vol.31,No.3, pp.405-413, 1990.3
[3] 金子ほか：「表記変換つきの形態素解析プログラムとその応用」，情報処理学会第45回全国大会 4C-4, 1992.10