72

印刷漢字の特徴抽出

元岡 達 (東大) 田中英彦 (東大) 鈴木達郎 (東大)

§1 はじめに

我国において、印刷された漢字を、そのまま入力として扱うことは、情報処理技術の対象の拡大に欠くことのできない技術である。

しかし現在かならずしも実用化されていない原因としては、① 漢字の種類はアルファベットなどと比較して、桁違いに数が多いこと。② 漢字の各々の字画が多く、複雑な図形であること。などが挙げられる。

これを打破する為には、情報密度が高く、かつ安定度が十分高い特徴を、抽出する必要がある。

本論文では、以上のことを踏まえて我々の研究室で従来から行なって来た方法に改良を加えた特徴抽出法を述べかつ、それをもとにした認識方法についてふれたい。

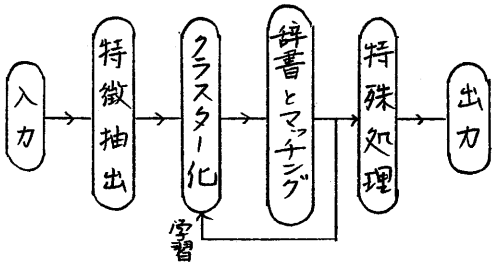

図1. 認識システム

§2 特徴抽出

漢字を一次元センサで縦に走査し、その上下の2本の線素の関係を次の二つに着目してコード化する。

1. 分割数の変化
2. 長さの変化(閾値を設ける)

これらのコードを、字の形のグラフ構造に対応させて、plex構造で表現する。(plex構造の各節には、座標、横幅も保存しておく。)

上の1,2の変化がない場合、コード化はされず、情報量は圧縮される。

このplex構造を用いて、一次元のコード列を作成する。(グラフ構造から、一意的に定まる。)

新しく長さの変化に閾値を設けることで、従来の方法よりも平均12倍以上の圧縮率とコードの安定化を得た。

また始点などの安定化を計る為、縦方向にも閾値を設け、その範囲内の変化は、まとめて一つのコードとした。

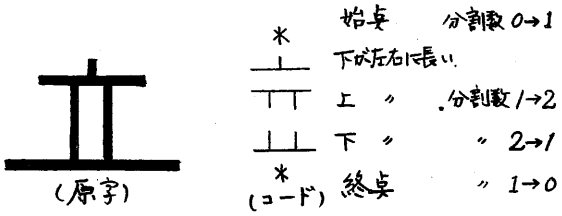

図2 コード化の例

§3 クラスター化

§2で抽出された特徴をもとに、いくつかのクラスターに粗分類する。

このクラスター化に必要な性格は、

1. 安定度(精度)が極めて高いこと
2. 分割効率がよいこと。

の二つが重要であるが、特に1.の性質は認識率に及ぼす影響からも不可欠である。

具体的なクラスター化の基準として

- 始点コードの数
- コード系列の長さ
- 各コードの横幅の分布(三段階)

等を用いることを考えており、各種の安定度が高く、クラスター化に適した特徴を発見する研究を続けている。

図3にその一例を示す。

このクラスター化の効果としては、

1. マッチングの回数を減らす。
2. マッチングの欠点を補う。

の二つが挙げられる。

ITEM	0	1	2	3	4	5	6	7	8	9	10	
TUP#	0	31	24	20	12	7	1	3	1	0	1	漢字100字の分布
C-LENG/5	1	18	24	22	20	9	4	1	1	0	0	
W50-50	46	22	16	5	3	2	3	0	1	0	1	
W50-70	47	20	19	11	2	1	0	0	0	0	0	
W70---	33	25	21	12	5	2	0	1	1	0	0	
TUP#	0	0	0	0	0	0	1	9	0	0	0	漢字の例「池」10個の分布
C-LENG/5	0	0	0	0	0	10	0	0	0	0	0	
W50-50	0	0	0	3	7	0	0	0	0	0	0	
W50-70	0	0	10	0	0	0	0	0	0	0	0	
W70---	10	0	0	0	0	0	0	0	0	0	0	

図3.

§4 マッチング

§2で得られたコード系列を用いてこれを別途に作成しておいた辞書の同一クラスター内のものと比較する。

<比較の方法>

両者の最大共通部分列（Largest Common Subsequence）即ち、どちらのコード系列の中にも順序をくずさずに含まれているコード系列のうち最大のものを求め，その長さを類似度として用いる。

この方法は、コードの出現順序を考慮に入れるので、情報量は極めて大きい。

§5 特殊処理

上記の分類，マッチングはすべての特徴を一様に扱って来たが、現実の漢字の中には全体的には相互によく似ていて、一部分のみが異なるものが多い。

この様なものは、場合に応じた分離方法で区別することが、結局能率を高めると思われる。

§6 各段階の情報量の考察

○ クラスター化の時

ここで分類まちがいすると，あとで取返しがつかないので、ここの精度は目標認識率を十分上回ってなければならない。例えば目標認識率を95％とし、5段階のクラスター化を行う場合、各段階の誤分類率をαとすると，

$$0.95 = (1-\alpha)^5 \fallingdotseq 1-5\alpha$$

$$\therefore \alpha = 0.01$$

従って99％の精度が要求される。

一般にn段階のクラスター化を行う場合、目標の$1/n$の誤り率が要求される。

精度を上げる為には，分割効率を犠牲にする方法が考えられる。たとえば§3で述べた5つの方法で各々平均5種類に分割するとし、高い精度を得る為に両隣りのクラスターもあわせて考えることにすると、分割効率は、

$$(1/5)^5 \div (3/5)^5 \fallingdotseq 1/250$$

程度になってしまう。

クラスター化の段階に学習機能を付加し、同一文字を複数のクラスターに属させることで、認識率を高める予定である。

○ マッチングの時

コード列の長さ---- l （平均20）

コードの種類----- n （実効値10）

とおけば、全体でn^lの可能性がある。

ある字のコード列が、標準の形（辞書の形）からr個以内の乱れに収まる場合その中に入るコード列の数は，

$$\sum_{k=0}^{r} n^k {}_lC_k \quad \cdots\cdots\cdots\cdots (1)$$

また、コード各々の安定度をpとすると、そのうちk個変化する確率は

$p^{l-k}\cdot(1-p)^k\cdot {}_lC_k$ になるから

r個以内の乱れに収まる確率は，

$$\sum_{k=0}^{r} p^{l-k}\cdot(1-p)^k\cdot {}_lC_k \quad \cdots\cdots (2)$$

$p=0.8$ と仮定すると，

(2)式の値が95％を越えるのは，$r=8$．

この時(1)式の値は1.8×10^{13}になるので，全体の10^{20}との比は5.5×10^{6}になる。

即ち形式的には500万に分けることになるが、実際にはかなりの片寄りがあるだろう。

またr個以内に複数の文字が入った場合には，特殊処理へ回されることになる。