

手書き文字認識における大分類のための決定木生成法の提案

篠 沢 佳 久[†] 大 駒 誠 一[†]

大規模な文字認識システムを構築する場合まず文字パターンがどのグループに属するのをおおまかに分類し、次に分類された各グループ内で詳細に認識し文字候補を1つに絞り込むという2段階による認識方法が一般的となっている。しかし前段の大分類部において1回で文字パターンを複数のグループに分類することは分類率の低下に直結するため逐次文字パターン群を分類し、決定木を生成する階層型の分類法も多く用いられている。決定木を生成していく段階では各層において分類すべき文字パターン群が前段階の分類の結果次第で変わってくるので、分類率を向上させるためには文字パターン群にとって有効な特徴を用いて分類すべきである。そのため各層ごとで複数の特徴の中からどの特徴をどのような基準で選択すべきかが問題となる。ここでの基準とは文字パターンを分割する時点での有効性だけでなく決定木全体を構築した際の有効性の尺度でなくてはならない。そこで本論文では、各層ごとにおいて文字パターンを分割する際、分類率だけでなく木のバランスなども評価する統合指標を用いて複数の特徴の中から最適な特徴を選択し、決定木を生成していく方法を提案する。そしてこの統合指標が最も高い特徴によって逐次最適な分類を行いETL9Bを対象として決定木を生成した結果、分類率において優れていることを示すことができた。

The Construction of Decision Tree for Rough Classification in Handwritten Character Recognition

YOSHIHISA SHINOZAWA[†] and SEICHI OKOMA[†]

Generally, a character recognition system classifies character patterns roughly in a first step and then recognizes them precisely. But such first rough classification is quite difficult in terms of generating classification boundaries. Therefore, our approach includes not dividing character patterns into many categories at one time but to classify them into a few categories step by step and to construct a decision tree in order to improve the classification rate. In this paper we propose how to construct a decision tree for rough classification. At each node of the decision tree, the groups of character patterns are different. If the group of character patterns changes, the feature that is suitable to classifying them changes too. The group of character patterns should be classified by the best feature. When we classify character patterns into a few categories, we use a total index to select the best from many features. The total index evaluates not only the classification rate but also the balance of the decision tree. In this work we constructed a decision tree for ETL9B. The results obtained from our work show an increased classification rate of character patterns.

1. ま え が き

手書き文字認識は古くから研究がさかんに行われており、今日に至るまでさまざまな認識方法が考案されてきた。字体が書き手によって異なり、筆順を問わない手書き文字認識の研究はパターン認識の分野において最も困難な課題の1つである。特に我が国において使用されている文字はJIS規格の文字だけでも6,349個にも及ぶため未知文字パターンの特徴ベクトルとすべての文字の標準文字パターンの特徴ベクトルとの距離

を1つ1つ求め、最も距離の短いものを認識結果とする従来の手法は計算量の点から困難である。そこで近年ではまずはじめに文字パターンをおおまかにいくつかのグループに分類し文字候補数を絞り込み(大分類)、そして次に絞り込まれたグループ内で詳細に認識する(詳細認識)2段階による認識方法がとられている^{1)~5)}。

ところが前段の大分類において1回の処理でいくつかのグループに分類することは結局のところ複雑な分類問題となり分類率の低下につながる。大分類での誤りはフィードバックが不可能なため深刻である。そこでゆるやかに分類し誤りが起きないように、文字パターンを徐々に分割し決定木(分類木)を生成していく階層

[†] 慶應義塾大学理工学部管理工学科

Department of Administration and Engineering,
Faculty of Science and Technology, Keio University

型の認識および分類方法も多く用いられている^{6)~7)}。

そこで筆者らは文字認識における大分類部の階層化を行い分類率の向上を試みた^{8)~9)}。文字パターンを分類していくうえでは2つの技術的な要素が問題となる。すなわちどのような分類方法により、どのような特徴を用いて分類していくのかということである。

分類方法については大分類において多くの研究者が線形分離を基本とした方法で行っている。これは大分類部はあくまでも文字候補を1つに絞り込むのではなく文字候補数を減らすことが目的だからである。文字パターンの特徴空間をおおまかに線形に分離できるところまでしていけばよいからである。

一方、特徴についてはこれまで多くの研究者によって数多くの特徴が提案されており、それらを用いる場合経験的要素を必要とする面もある。筆者らが大分類の階層化を図るうえで使用した文字パターンの特徴とその分類結果について、いくつか気づいた点があった。まず同じ文字パターン群を分類の対象としても、使用する特徴が異なれば分類結果が異なるということである。すなわち文字パターン群が変わればその文字パターン群を分類しやすい特徴も変わるということである。また文字パターン群を同一の特徴を用いて分類し続けていくと、徐々に分類しにくくなる傾向がある。これは同一の特徴を用いて分類し続けていくと、同じ特徴を持った文字パターンが同一グループに集中することになるからである。大分類部を階層化するうえで最も問題となるのが各層ごとで分類すべき文字パターン群が前段階の分け方次第で動的に変化することである。こうしたことから結論づけられることは分類率の向上のため各層ごとで使用する特徴は、その段階で分類すべき文字パターン群にとって最も有効なものを使って分類するように動的に変えるべきである。またそのような設計に基づいて決定木を構築していくのであれば、各層で使用される特徴をどのような基準(評価関数)によってどのように自動的に選択できるようにするかということが問題となる。

ここでの基準というのは分割時での有効性を図る尺度としてだけでなく階層化を図るうえで全体的な分類率を向上させるための要因とすべきである。決定木を生成していく方法は機械学習 ID3¹⁰⁾などさまざまあるが、その評価は葉グループを派生する時点に行い、その時点での分割情報しか利用できないため、生成される決定木にさまざまな問題が生じる。次章以降に詳細に述べるが、分類率のみを唯一の基準とした場合、生成した決定木のバランスにも問題が生じることがある。木のバランスが悪いということは木の末端の葉に

至るまでの経路にばらつきがあることを意味する。すなわち計算量の増大のみならず分類率の低下につながる。またバランスの悪い決定木を用いて実際に未学習文字パターンを分類させてみると、ある特定の文字パターンだけが非常に分類に時間がかかったり、もしくは非常に分類率が悪いという結果が起きてしまうなどの問題が生じる。

そこで本論文では文字認識の大分類部において階層化を図るうえで、各層で分類する文字パターン群に対して特徴を変化させ、最も有効な特徴を使用することによって分類率の向上を図る。そうした際に各層でどのような基準を設けて特徴の自動選択を行うかを論じ、大分類のための決定木生成法を提案する。

2. 文字パターンの分類方法

2.1 文字パターンのグループ分け

本論文ではまず文字パターンのグループ分けを決めて、次にそのグループ分けに従って文字パターンを分類できる識別機械を作成するという2段階の方法を用いた^{8)~9)}。文字パターンの分類方法については数多くの手法が提案されているが本論文では Kohonen の自己組織化アルゴリズム¹¹⁾を用いた。Kohonen の自己組織化アルゴリズムは以下のとおりである。

文字パターンが P 個 ($i = 1, 2, \dots, P$)、文字パターン i の特徴ベクトルを $\mathbf{x}_i^t = (x_{i1}, x_{i2}, \dots, x_{iN})$ (N : 特徴ベクトルの次元数) とする。特徴空間上に M 個 ($j = 1, 2, \dots, M$) の参照ベクトルがあり、 j 個目の参照ベクトルを $\mathbf{v}_j^t = (v_{j1}, v_{j2}, \dots, v_{jN})$ とするとき、以下のように参照ベクトルを修正する。

- (1) 学習回数を t とすると $t = 0$ において $\mathbf{v}_j(t)$ を乱数化する。特徴ベクトルを順次入力し(2)、(3)を参照ベクトルが安定するまで繰り返す。
- (2) 入力した特徴ベクトル \mathbf{x}_i と最も距離の短い参照ベクトル \mathbf{v}_s を求める。

$$t = t + 1$$

$$\|\mathbf{x}_i - \mathbf{v}_s(t)\| = \min_j \|\mathbf{x}_i - \mathbf{v}_j(t)\|$$

- (3) 入力した特徴ベクトルと最も距離の短い参照ベクトル $\mathbf{v}_s(t)$ をその特徴ベクトルの方向に多少修正する。該当しない参照ベクトルはそのままである。

$$\begin{aligned} \mathbf{v}_j(t+1) \\ = \mathbf{v}_j(t) + \alpha(t) \cdot (\mathbf{x}_i - \mathbf{v}_j(t)) \end{aligned}$$

$$\text{for } j = s$$

$$\mathbf{v}_j(t+1) = \mathbf{v}_j(t)$$

$$\text{for } j \neq s$$

表1 Kohonen の自己組織化による文字パターンの分割
Table 1 Rough classification of character patterns using Kohonen's self-organization.

文字	グループ 1	グループ 2	グループ 3	分け方
あ	90	7	3	グループ 1
い	16	72	12	未判定
う	4	4	92	グループ 3
え	95	0	5	グループ 1
お	88	11	1	未判定
か	16	30	54	未判定
き	23	1	76	未判定
く	0	3	97	グループ 3
け	0	90	10	グループ 2
こ	5	76	19	未判定

$$\alpha(t) = 0.1 \times \left(1.0 - \frac{t}{\text{最大学習回数}}\right)$$

α は学習係数であり学習回数が増えるにつれ小さくしていく。本論文では初期状態は 0.1 としている。この方法によって類似した特徴ベクトルに対しては同一の参照ベクトルによって代表されるようになる。すなわちそれぞれの参照ベクトルは各グループを代表するベクトルになり、その参照ベクトルによって類似した特徴ベクトルごとにグループとして分割できるようになる。

ところが手書き文字のような変形が激しいパターンにおいては、同じ文字パターンから同一の特徴を抽出することは難しい。そのため同じ文字パターンが同じグループとして分割されるとは限らない。たとえば平仮名 10 文字、1 文字につき 100 個を学習文字パターンとして使い、ある任意の特徴を用いて、上記のアルゴリズムにより 3 つのグループに分割した結果を例として表 1 に示す。

表 1 の『か』のように全部のグループに分割されてしまった場合、どのグループに属しているか判断しかねる。そこで明確に分類できている文字だけを分類できたと判定する。判定方法は、

$$\frac{\text{最も多く分割された文字パターン数}}{\text{学習に使用した文字パターン数}} \geq \beta \quad (1)$$

とする。 β は分類判定のための係数であり、判定係数と呼ぶ。表 1 は $\beta = 0.9$ とした場合のグループ分けの結果である。分類結果の各文字パターンに対して式 (1) を用いて分類できているかを判定する。

文字パターン『あ』は 100 個の学習文字パターンのうち 90 個がグループ 1 に分類されている。したがって $\frac{90}{100} \geq 0.9$ より式 (1) が成り立っているのでグループ 1 に分類されるものと決める。一方、文字パターン『い』は 100 個の学習文字パターンのうち 72 個がグループ 2 に分類されているが、 $\frac{72}{100} \leq 0.9$ より式 (1)

表2 再学習による識別面の生成
Table 2 Construct of classification boundaries by relearning.

文字	グループ 1	グループ 2	グループ 3
あ	100	0	0
い	-	-	-
う	0	0	100
え	100	0	0
お	-	-	-
か	-	-	-
き	-	-	-
く	0	0	100
け	0	100	0
こ	-	-	-

を満たしていないので、ここで使用した特徴では分類できなかったと見なす。このような場合はグループ 1 からグループ 3 まですべてに属すると判定してしまう。このような判定をした表 1 のグループ分けの結果は、

- グループ 1… あえ (いおかきこ)
- グループ 2… け (いおかきこ)
- グループ 3… うく (いおかきこ)

となる。分類できなかった文字パターン群『いおかきこ』はすべてのグループに登録される。さらにグループ 1 からグループ 3 において他の特徴を用いてこのようなグループ分けを繰り返す。

2.2 識別面の生成

表 1 において、文字パターン『あ』は 100 個の学習文字パターンのうち 90 個はグループ 1 にきちんと分類できているが、残り 10 個は分類できていない。そこで残りの 10 個もグループ 1 に分類できるようにする必要がある。

そのため表 2 に示すような分類が可能な識別機械を新たに構築しなければならない。この識別機械についてはニューラルネットワークの 1 つであるバックプロパゲーションネットワーク (以下、BP ネットと略す) を用いて構築した¹²⁾。BP ネットは図 1 に示すように入力層、隠れ層、出力層から成る 3 層型のフィードフォワードネットワークである。入力層のニューロンは使用した特徴の個数分用意する。隠れ層のニューロンの個数は任意である。出力層のニューロンの個数は分類すべきグループ数分用意し、出力層の第 1 番目のニューロンはグループ 1 というように対応づけする。そして入力層の各ニューロンへは抽出した特徴ベクトルを入力し、グループ決めをしたおりに出力層のニューロンが発火するように学習させるために、外部から教師信号を与える。グループ 1 と対応した出力層のニューロンのみを発火させる場合の教師信号は $T = (1, 0, 0)$ である。この教師信号を用いてニューラ

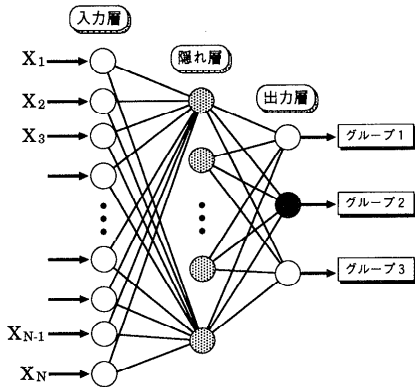


図1 再学習のためのBP ネット

Fig.1 Backpropagation network for relearning.

ルネットワークを誤差逆伝播アルゴリズム (BP) で学習し、適切な出力が出るようにする。グループを決定しなかった文字パターンは学習しない。

2.3 再学習の意味

この2段階による方法ではまず文字パターンの特徴空間を Kohonen の自己組織化によって線形分離し、おおまかに文字パターンをいくつかのグループに分けてしまう。その際生成された識別面近傍にある文字パターンは、今回利用した特徴では分類できなかったと見なし分類不可能と判定する。次に今回分類できた文字パターンに対してはBP ネットを用いて新たに詳細に識別面を生成し、前段階で決められたグループ分けができるようにする。

すなわち本論文の文字パターンの分類方法は、まず分類する文字パターンがある特徴空間の中でどのように分布しているかを調べ、そして局所的に集中している文字パターン群のみを分類できるものと判定し、大域的に散らばっている文字パターン群は無視し、分類可能と判定した文字パターン群に対して識別面を生成するという方法である。そして分類不可能と判定された文字パターン群については今回使用したものは別な特徴を用いて同様に分類する。

3. 分類木の生成

3.1 特徴選択の必要性

分類木を生成していくうえで問題となるのは各層ごとで動的に変化していく文字パターン群に対して最適な特徴をどのような基準で選択していくかということである。単一の特徴のみを利用して木を生成すれば同じ特徴を持った文字パターンが同じグループに集まることになり、徐々に分類しにくくなる傾向がある。そこであらかじめ使用すると決めておいた複数の特徴の

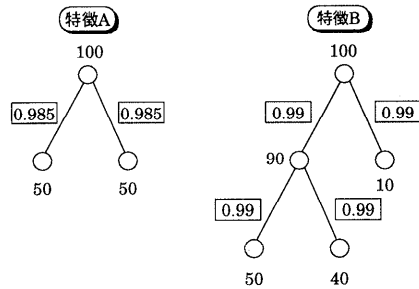


図2 分類木の局所生成における問題

Fig.2 The problem concerning local construct of decision tree.

中からどの特徴を選択するべきか、分類結果を評価し決定しなければならない。次にどのような基準 (評価指標) を用いて特徴を選択するべきかを論じる。

3.2 分類度を用いた特徴選択

文字認識で最も重要な指標は分類率と認識率である。そこで Kohonen の自己組織化によって分類した結果、文字パターンがどの程度適切に各グループに分類できたかを示す値として分類度という指標を導入する。分類度は式 (1) を満たし分類できたと判断した学習文字パターンの分類結果から求める。

$$\text{分類度} = \sum_{i=1}^P X_i \times \frac{\text{パターン } i \text{ における正当数}}{\text{学習文字パターン数}}$$

$$X_i = \begin{cases} 1 & (\text{グループを決定した場合}) \\ 0 & (\text{未判定とした場合}) \end{cases}$$

分類度は1に近いほど、分類に使用した特徴が優れていることを意味している。文字認識においてはこの分類度のみ重点を置けばよいと考えがちであるが、分類木の生成は逐次局所的なので決定木のある任意のノードにおける分類度を最大にして生成していったからといって全体的な木の分類率が高くなるとは必ずしもいえない。

図2に示すように、たとえば100個の文字パターンがある任意の特徴を仮に特徴Aと呼ぶことにすると、その特徴Aでは50個、50個に分割され、また別の特徴Bでは90個、10個に分割されたものとする。特徴Aは特徴Bと比較して均等に文字パターンを分類してしまっているため、一般的に分類度は特徴Bの方が高くなる傾向がある (図中の枠の中に示した数値が分類度)。しかし、特徴Bで生成した分類木では葉グループで処理するパターン数が特徴Aの場合と比較して多いので、さらに90個をもう1回分類する必要がある。すると全体的な分類度は、はじめ

に特徴 B を使用した場合よりも特徴 A を使用した場合の方が高くなる。またははじめに特徴 B を用いて生成したような木の場合、バランスがとれていないため次の詳細認識部で処理しなければならない文字パターン数がまちまちになってしまい、均等な能力を持つ詳細認識部を次に置けばよいということでは済まされなくなってしまう。

このように局所的に分類度が高くなるように分類しても必ずしも総合的に分類木全体の分類度が良くなるとは限らない。局所的に分割していく際には総合的なバランスも考慮した指標を用いるべきである。

3.3 エントロピーを用いた特徴選択

どの特徴を使用するか分類結果を評価し分類木を生成していく方法として一般的な手法の1つに **ID3** がある。ID3 ではこのような評価にエントロピーを用いている。エントロピーは分類の良さを示す指標だけでなく分類した際、バランスも考慮した指標である。

エントロピーの求め方は以下のとおりである¹⁰⁾。事例の集合 S からランダムに1つの事例を選択し、集合 S に属する事例数を $|S|$ を表し、それがグループ C_j に属している確率を、

$$\frac{\text{freq}(C_j, S)}{|S|}$$

とするとき、その情報量は、

$$-\log_2 \left(\frac{\text{freq}(C_j, S)}{|S|} \right)$$

である。そして S 内での頻度の重み付けを行いグループ全体に対しての平均を求めると、

$$\begin{aligned} \text{info}(S) = & - \sum_{j=1}^k \frac{\text{freq}(C_j, S)}{|S|} \\ & \times \log_2 \left(\frac{\text{freq}(C_j, S)}{|S|} \right) \end{aligned}$$

となる。これが集合 S のエントロピーと呼ばれる量である。これを学習文字パターンの集合 T に適用する。学習文字パターンの集合 T を特徴 A で n 個のグループに分類した結果について、各グループ内のエントロピー $\text{info}(T_i)$ を求める。その後で荷重平均をとって、

$$\text{info}_A(T) = \sum_{j=1}^n \frac{|T_j|}{|T|} \times \text{info}(T_j)$$

となる。そして分類前の集合 T のエントロピー $\text{info}(T)$ との差、すなわち

$$\text{gain}(A) = \text{info}(T) - \text{info}_A(T)$$

が最大になるような特徴を選択する。実際には式 (1)

式を満たした文字パターン群からエントロピーを求める。

3.4 統合指標を用いた特徴選択

3.4.1 線形分離度の導入

本論文の文字パターンの分類方法では分類できたもの、すなわち式 (1) を満たした文字パターンと、満たさず分類できなかった未判定文字パターンに分かれることになる。エントロピーは未判定文字パターン以外から求めることになるが、そのため次のような問題点が生じる。

図3に示すように100個の文字パターンを特徴 A では70個、60個に、特徴 B では90個、90個に分かれたものとする（括弧内の数字は式 (1) を満たした文字パターン数）。エントロピーによる結果はバランスを重視するため、特徴 B を用いた結果を高く評価する。そのため特徴 B を用いた結果は特徴 A を用いた方よりも葉グループで次に処理しなければならない文字パターン数が多くなってしまふ。これは未判定とされた文字パターンを無条件に各グループに付加させているためである。エントロピーのみを特徴選択の指標とした場合、バランスを重視するあまり、木の高さを無視してしまうことがある。これを解決するために線形分離度という指標を導入する。

線形分離度は Kohonen の自己組織化によって文字パターンの特徴空間をいかに線形に分類しているかを示す指標であり、以下の式から求めることにする。

$$\text{線形分離度} = \frac{\text{式 (1) を満たした文字パターン数}}{\text{文字パターン数}}$$

線形分離度は1に近いほど使用した特徴が文字パターン群を分割するのに優れているものと判断する。表1に示した分類例の場合、式 (1) を満たし分類できた文字パターン数は5個なので、線形分離度 $= 1 - \frac{5}{10} = 0.5$ である。

すなわち線形分離度が高い特徴を選択するようにすれば葉グループで次に処理（詳細認識）しなければならない文字パターン数が多くなってしまふという問題

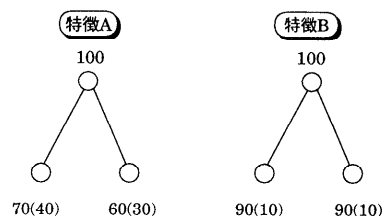


図3 エントロピーを用いた特徴選択における問題
Fig. 3 The problem in terms of using entropy on selecting feature.

点をさけることができるようになる。

3.4.2 木のバランス

エントロピーと線形分離度をどのように組み合わせるかということが問題となる。エントロピーは分類度と木のバランスをともに考慮している指標であるとはいつても両者の違いがはっきりとしない。そこで本論文では、エントロピーと線形分離度を組み合わせるのではなく、分類度、木のバランス、線形分離度を組み合わせた統合指標を用いて特徴を選択する。分類度および線形分離度はこれまでとおりであるが、木のバランスは別式を用いて評価することにした。分類木は局所的に生成していくので分割時のノードでしかバランスを評価することができない。各グループに分類されたうちの最大文字パターン数を P_{\max} 、また最小文字パターン数を P_{\min} とし、以下の2つの指標を用いた。

$$\text{バランス 1} = 1 - \frac{P_{\max}}{\text{文字パターン数}}$$

$$\text{バランス 2} = 1 - \frac{P_{\max} - P_{\min}}{\text{文字パターン数}}$$

2つの指標とも1に近いほど使用した特徴が文字パターン群を分割するのに優れているものと判断する。本論文による分割方法では未判定文字パターンは無視され、各グループに登録してしまうので、各グループ中に分類されたうちの最大文字パターン数が少ない方が優れた分類方法であると見なす。その指標がバランス1である。バランス2は各グループで次に分類しなければならない文字パターン数のばらつきを表し、均等に分割しているかを評価する指標である。

3.4.3 指標の統合化

文字パターン群の分割を評価するのに分類度、線形分離度、バランス、以上3つの指標を用いるがこれらの指標は複雑に関連し合っている。そこでこれらの指標を統合化した評価指標（統合指標）を使う。統合指標は次式のようにこれら3つの指標を合計したものである。

$$\text{統合指標} = \text{分類度} + \text{線形分離度} + \text{バランス 1} + \text{バランス 2} \quad (2)$$

この統合指標を最大にする特徴をその文字パターン群を分割するのに最適な特徴として選択する。

3.5 分類木生成アルゴリズム

式(2)を用いて文字パターン群を分割する特徴、また拡張して分割数も選択し、分類木を生成していく。分類木生成アルゴリズムは以下のとおりである。

ステップ1

使用する特徴を T_i ($i = 1 \dots N$)、分割数を

```

c (c = 1...C) とするとき,
for ( i = 1 ; i <= N ; i++ ) {
    for ( c = 1 ; c <= C ; c++ ) {
        Kohonen の自己組織化 (  $T_i$  , c ) ;
        統合指標 (  $T_i$  , c ) ;
    }
}

```

というように特徴、分割数を逐次変え、Kohonenの自己組織化を行い文字パターン群を分割し、式(2)によって統合指標をそれぞれ求める。

ステップ2

ステップ1で求めた統合指標を最大にする特徴 T_{\max} 、分割数 c_{\max} をこの文字パターン群を分割するのに最適な特徴、分割数とする。

ステップ3

決定した特徴 T_{\max} 、分割数 c_{\max} でBPにより再学習を行う。

ステップ4

分割した各グループの文字パターン数が M 個未満かどうかを調べ、 M 個以上ならばその文字パターン群に対してステップ1からステップ3まで繰り返す。 M 個未満ならば、そこで葉の生成を停止する。

4. 評価実験

4.1 使用した手書き文字パターン

本論文で提案する方法により分類木を生成し評価実験を行う。実験は3種類行った。まず予備実験として本論文で提案する統合指標が分類木を生成していくうえで有効な指標かどうかを調べる（実験1および実験2）。そして次に本論文の提案する分類木生成方法により大規模な文字の大分類が可能かどうかを調べる（実験3）。

本論文では通産省電子技術総合研究所提供の文字データベース ETL8B および 9 を使用した。実験1および実験2では ETL8B 中の最初の1セット 320文字を使用した。1文字につき50パターンを学習文字パターンとして手順に従って分類木を生成し、また別の50パターンを実験用の未学習文字パターンとして扱うことにした。

また実験3では ETL9、最初の1セット 3,036文字を使用した。1文字につき30パターンを学習文字パターンとして用い分類木を生成し、残り10パターンを実験用の未学習文字パターンとして扱うことにした。

元の文字画像にはノイズが含まれているため、前処理を施す必要がある。そこで

- ノイズ除去処理
- 大きさの正規化処理
- マスクパターンを用いた細線化処理¹³⁾

といった前処理を行った。その前処理を施した2値画像から以下の特徴を抽出した^{14)~15)}。

- メッシュ特徴 (256 次元)
- 端点交点特徴 (256 次元)
- 方向線素特徴 (400 次元)
- 輪郭線特徴 (169 次元)
- 周辺分布特徴 (96 次元)
- ペリフェラルパターン特徴 (192 次元)

4.2 実験1—バランスのとれた分類木の作成

4.2.1 Kohonenの自己組織化によるグループ分け

まず実験1として3章で述べた3つの指標を用いて分類木を生成し、どの評価指標が特徴選択に有効かを調べる。予備実験より今回使用したKohonenの自己組織化アルゴリズムでは分割数を増やせば各指標は低下することが分かったので、逐次分割していくグループ数は2個と固定した。木の高さは6とし、分割するグループ数は32としたバランス木を作成した。式(1)で使用する判定係数は1.0に近づけすぎると分類しにくくなるので $\beta = 0.9$ とした。学習回数は300回とした。その結果を表3、表4に示す。表3に提示した情報は32個の各グループに分類された文字パターン数の平均、その標準偏差、各グループに分類されたうちの最大、最小文字パターン数である。表4には分類木を生成するうえにおいて選択された特徴の回数を示す。分類木を作成する際には31回の特徴選択が行われる。

表3より分類度のみを特徴選択の指標として使用した場合は予想したとおり各グループごとでのばらつきが生じ、最大文字パターン数は191個と階層化した効果がまったく現れていない。

また表4より、他の2つの指標は特に固定した特徴ではなく、さまざまな特徴を使用しているが、分類度の場合固定した特徴、特に輪郭線特徴を使用する傾向があることが分かる。1つの特徴で分類木を生成していくとバランスに問題が生じることがこのことから明らかである。このように分類度のみを特徴選択の指標として使用した場合は他の2つと比較して候補文字を絞ることが困難であることが分かる。

エントロピーと統合指標の場合は類似した結果が出ている。今回分類木を生成するためには31回特徴を選択することになるが、そのうち22回はエントロピーと統合指標によって選択された特徴が一致した。残りの9回は図3で示した問題点が生じていた。その場合

表3 各指標を用いて生成した分類木

Table 3 The result of decision tree constructed with each index.

指標	平均	標準偏差	最大	最小
分類度	37.0	44.63	191	5
エントロピー	66.6	26.34	106	23
統合指標	58.8	25.18	97	26

表4 各指標ごとの特徴の使用回数

Table 4 Number of times of feature used for each index.

使用した特徴	分類度	エントロピー	統合指標
メッシュ	8	8	5
端点交点	0	4	3
方向線素	3	5	7
輪郭線	13	2	5
周辺分布	3	5	3
ペリフェラル	4	7	8

予想したとおり線形分離度を導入した統合指標ではバランスよりも線形分離度が高くなる分類結果を採用していた。その結果、表3からエントロピーと統合指標では各グループごとの文字パターン数のばらつき(標準偏差)は同じなのに統合指標の方が平均文字パターン数および最大文字パターン数が小さくなるという効果が表れ、統合指標を用いた場合、平均文字パターン数および最大文字パターン数はエントロピーを用いた場合の90%程度まで減らすことができた。

4.2.2 BPによって再学習した分類木

次に生成した分類木に対してBPを用いて再学習させた。BPはモーメント法¹²⁾を用いた。モーメント法を行う場合に使用する2つの係数、学習係数は0.5、モーメント係数は0.9とした。学習回数はすべて100回と固定した。BPネットの隠れ層の数は学習する特徴ベクトルの次元数とは関係なく予備実験より16個と固定した。

末端の葉のグループからの出力値はその葉に至るまでのBPネットからの出力値の合計とした。すなわち31個のBPネットからの出力値を求めている。木の各ノードごとに再学習し統合化したニューラルネットワークに対して未学習文字パターンを大分類させた。第5候補までの累積分類率の結果を表5に示す。分類率とは適切にグループ分けができた文字パターン数の合計を調べた文字パターン数の合計で割った値のことである。

これら3つの指標を比較するとどれも同程度の分類率だが、表3と比較してみると文字候補数を一番絞り込むことのできる統合指標が特徴選択において最も有効な指標であることを示すことができた。

表5 バランスのとれた分類木の分類率

Table 5 The classification rate of balanced decision tree.

指標	分類度	エントロピー	統合指標
第1候補	95.2%	95.2%	95.3%
第2候補	97.9%	98.1%	98.1%
第3候補	98.9%	99.1%	99.0%
第4候補	99.2%	99.4%	99.4%
第5候補	99.4%	99.5%	99.5%

表6 末端の葉の文字パターン数の上限を決めた分類木

Table 6 The result of unbalanced decision tree.

指標	最短ノード数	最長ノード数	分類率
分類度	3	16	94.2%
エントロピー	4	10	95.7%
統合指標	4	8	96.1%

4.3 実験2—木の高さを制限しない分類木の作成
 実験2として次にバランスのとれた木ではなく末端の葉グループで処理する文字パターン数が M 個以下になるまで分類し続ける方式で分類木を生成し実験1と同様、統合指標の有効性を確認する。分類木を生成する条件はこれまでとおりとした。末端の葉で処理する文字パターン数の上限は70個とした。末端の葉のグループからの出力値はその葉に至るまでのBPネットからの出力値の合計をこれまで通過してきたノード数で割った値とした。その結果を表6に示す。表6には生成した分類木で末端の葉までの最短ノード数、最長ノード数、未学習文字パターンの分類率を示した。

表6より分類度のみを特徴選択の指標とした場合、非常にバランスの悪い木となってしまう、極端に長い枝ができてしまう。BPにより再学習しているからといって局所的に分類するときの分類率が100%にならない限り木の高さが高くなるほど分類木全体の分類率は低下してしまうことは明らかである。統合指標を特徴選択の指標に用いた場合、分類率が96.1%と最も良い結果となったが、これは全体的な木の高さを低くすることができたからである。実験1および実験2の結果より分類木生成において逐次文字パターンを分類する際、分類度だけでなく木のバランス、線形分離度といった局所的にではなく全体的に木を評価できる指標を特徴選択の基準として用いた方が総合的な分類率では優れていることを示すことができた。

4.4 実験3—大規模な文字の大分類

実験3として本論文で提案する分類方法により大規模な文字の大分類が可能かどうかETL9(3,036文字)を用いて実験を行う。特徴選択に使用する指標は2つの実験によりその有効性を示した統合指標を使う。木の高さは8とし、分割するグループ数は128個とする

表7 ETL9の大分類の結果

Table 7 The result of decision tree constructed for ETL9.

	平均	最大	最小	分類率
メッシュ	242.3	412	63	84.2%
端点交点	427.3	619	265	78.6%
方向線素	281.5	409	58	86.3%
輪郭線	192.5	406	40	86.8%
周辺分布	321.5	525	102	80.5%
ペリフェラル	185.6	434	78	87.6%
分類木	145.3	215	103	93.2%

バランス木を生成した。使用する特徴は分類木の高さ4まではメッシュ、方向線素、輪郭線、ペリフェラルパターンの4種類から選択し、それ以降は6種類すべての特徴から選択させることにした。Kohonenの自己組織化においては判定係数は $\beta = 0.9$ 、学習回数は200回とした。再学習のために使用するBPネットの隠れ層の数は予備実験より高さ4までが96個、それ以降は32個と固定した。実験2と同様にBPはモーメント法を用いた。モーメント法を行う場合に使用する2つの係数、学習係数は0.5、モーメント係数は0.9とした。学習回数はすべて100回と固定した。

また比較として2章で述べたKohonenの自己組織化により6つの特徴を用いて大分類を行った。分割するグループ数は分類木と同様に128個とした。判定係数はこれまで同様 $\beta = 0.9$ とし、学習回数は特徴の次元数とは関係なくどれも200回とした。

末端の葉のグループからの出力値はその葉に至るまでのBPネットからの出力値の合計とした。すなわち127個のBPネットからの出力値を求めている。

その結果を表7~9に示す。表7で提示した情報は128個の各グループに分類された文字パターン数の平均、各グループに分類されたうちの最大、最小文字パターン数、そして未学習文字パターンに対する分類率である。表8では分類木を生成するうえにおいて選択された特徴の使用回数を示す。表9では第5候補までの累積分類率を示す。

表7から階層化しなかった場合と階層化した場合の差がはっきりと現れている。各グループごとに分類された文字パターン数においては階層化した場合145.3個と、階層化しなかった場合最もその平均文字パターン数の少ないペリフェラルパターン特徴を用いた場合の185.6個と比較しても約78%程度まで減らすことができた。また文字パターン数のばらつきにおいても階層化しなかった場合最も少ない最大文字パターン数は輪郭線特徴の406個と比較して、階層化した場合は215個と約半分まで減らすことができた。特に分類率については階層化しなかった場合最も高いペリフェ

表 8 ETL9 の大分類における特徴の使用回数

Table 8 Number of times of feature used on decision tree for ETL9.

指標	メッシュ	端点交点	方向線素	輪郭線	周辺分布	ペリフェラル
高さ 4 まで	5	—	3	3	—	4
それ以降	21	9	25	19	12	26

表 9 ETL9 の大分類の累積分類率

Table 9 The classification rate of decision tree for ETL9.

	ペリフェラル	分類木
第 1 候補	87.6%	93.2%
第 2 候補	90.3%	96.5%
第 3 候補	92.1%	97.8%
第 4 候補	92.9%	98.9%
第 5 候補	93.5%	99.1%

ラルパターン特徴の 87.6%と比較して階層化した場合は 93.2%と 5%程度向上させることができた。

ところですべての BP ネットからの出力値を求めるのではなく、その 2 つの出力値を比較し、大きい値を出すノードを選択する方式で分類していく場合、その分類率は 91.5%であった。この方法だと 7 回 BP ネットから出力値を求めるだけで済むが、分類率が低下するだけでなく、第 1 候補のみしか選択することができない。1 つ 1 つの BP ネットの未学習文字パターンに対する平均分類率は約 99%であった。分類率をさらに向上させるためにはこの点の改良が不可欠である。

表 8 には特徴の使用回数を示したが、端点交点特徴、周辺分布特徴を除いて他の 4 つの特徴はほぼ均等に使われている。表 7 の結果から明らかなように端点交点特徴、周辺分布特徴を使用した場合、他の 4 つの特徴と比較して分類結果が劣っている。そのため選択される回数が少なかったのである。このことから単独に分類を行いほぼ同じような精度の分類結果が得られるような特徴を特徴選択の候補とすべきである。

表 9 には表 7 より最も分類率の高いペリフェラルパターンを用いた場合と分類木の場合の第 5 候補までの未学習文字パターンに対する累積分類率を示した。分類木の場合第 5 候補まで求めた結果、分類率 99%まで達することができた。一方階層化しない場合、なかなか累積分類率が上昇しない。次の詳細認識において生成した分類木の場合、第 5 候補までのグループ内で認識を行えばよいが、階層化しない場合だとそれ以上の個数のグループにおいて認識を行わなければならない。

5. 考 察

5.1 分類率の向上の条件

本論文では文字認識の大分類において 1 回の分類でいくつものグループに分けることは分類率の低下につ

ながるということから議論し、そのため徐々に文字パターンを分類していく決定木の生成法を提案した。そして文字パターンを複数のグループに分類するのではなく 2 個のグループに分類していく二分木を提案した方法に基づき生成し、文字認識データベース ETL9 で実験評価を行った結果、提案方法の有効性を確認できたものの、第 1 候補の分類率を 100%まで向上させることはできなかった。これは前章でも述べたが、1 回ごとに分類するグループ数を 2 個と最低限に抑え、さらに教師型学習である BP で再学習させたにもかかわらず分類率は 100%には達しなかったためである。木の高さを N とし、BP ネットの平均分類率を R とすれば、生成した木の全体的な分類率は R^{N-1} である。BP ネットでの分類率 R を 100%にしない限り全体的な分類率も 100%にはならない。また木の高さ N が高くなればなるだけ全体的な分類率も低下してしまうが、この点においては統合指標を特徴選択の指標として用いれば木の高さを低くすることが可能であることを示した。そのため今後は BP ネットの分類率が 100%に達するように、より優れた特徴を使用したり BP ネットを改善したり個々の優れた技術を取り入れ分類率を高めていくつもりである。

5.2 計算量の比較

階層化した場合としない場合の計算量および統合指標を用いて生成した分類木の計算量の違いについて比較検討する。逐次最適な分割を行い階層化していった方が絞り込むことのできる文字候補の数に関しても分類率に関しても有利であることが分かった。これは 1 回で適切な識別面（識別関数）を生成するよりも逐次識別面を生成した方が、より精度の高い識別面を生成できることから明らかである。

しかし計算量に関しては階層化した場合不利である。今回評価実験で比較した Kohonen の自己組織化と本提案法とを比較してみる。使用する特徴ベクトルの次元数を T 個、分割数を O 個とする。階層化しない場合出力値を求めるのに必要な計算量はこの 2 つを掛け合わせた $T \times O$ に比例する。そしてさらに O 個の出力値をソートする必要がある。一方、階層化した場合、今回生成したようなバランスのとれた二分木であると仮定し、BP ネットの隠れ層の数を M 個とする。その場合出力値を求めるのに必要な計算量は、分類木のルー

トから $\log_2 O - 1$ の高さまでの各グループの BP ネットでの計算量の合計, $\sum_{i=0}^{\log_2 O - 1} 2^i \times (T * M + M * 2)$ に比例する. さらに階層化しない場合と同様に O 個の出力値をソートする必要がある.

実験結果から階層化した場合としなかった場合とでは絞り込むことができる文字候補数が異なり, 次の詳細認識部で処理しなければならないグループの個数が異なることから単純には比較できないが, 単純に考えると木構造化した場合 $\sum_{i=0}^{\log_2 O - 1} 2^i$ 倍, 出力値を求めるための計算量がかかってしまう.

このことは階層化した場合必ず生じる問題であり, 計算量を少なくするためにも木の高さを低くしなければならない. 今回特徴選択に使用した 3 つの指標の中で統合指標が最も木の高さを低くすることが可能であることを示した. 統合指標を特徴選択の指標にすることによって大分類部の階層化において生じる計算量の問題点を改善することができるのである.

今回は分類していくグループ数を最低限の 2 個としたが, 今後は Kohonen の自己組織化以外のクラスタリングによって分割数の影響を調べ, 分割数を増やすことによって木の高さを低くし計算量を削減していくつもりである.

6. むすび

本論文では文字認識における大分類の木構造化を試みた. まず Kohonen の自己組織化により文字パターンの特徴空間をおおまかに分割し, 個々の文字パターンがどのグループに属しているかを決めた. 次に前段階で決めたようにグループ分けのできる分類機械を BP を用いて作成した. この 2 段階による方法で逐次文字パターンを分類していく決定木を作成する.

大分類部を階層化していくうえで問題となるのが各層ごとで分類しなければならない文字パターン群が前段の分け方次第で動的に変わるということである. これまで多くの文字の特徴が提案されてきているが, 分類すべき文字パターンの集合が変われば有効な特徴も変わるので, 各層ごとで有効な特徴を選択しなければならない. そこでどの特徴で文字パターン群を分類すべきかという指標が必要となる. その特徴選択の指標として, 正確に文字パターンを分類しているか (分類度), 文字パターンを適切に線形に分離している割合が高いか (線形分離度), そして木を生成していくうえでのバランス, この 3 つの要素を考慮した統合指標を提案した. その結果この統合指標を用いて特徴選択を行わせ, 各層で用いる特徴を変化させ分類木を生成しその有効性を確認した.

謝辞 本研究を行う際に, 貴重な手書き文字データベースを提供してくださいました通産省工業技術院電子技術総合研究所の皆様にご感謝いたします.

参考文献

- 1) 岩田, 富麻, 松尾, 鈴木: 大規模 4 層ニューラルネットワーク CombNET, 電子情報通信学会論文誌 (D-II), Vol.J73-D-II, No.8, pp.1261-1267 (1990).
- 2) 堀田, 岩田, 松尾, 鈴木: 大規模ニューラルネットワーク CombNET-II, 電子情報通信学会論文誌 (D-II), Vol.J75-D-II, No.3, pp.545-553 (1992).
- 3) 猪野, 岩田: CombNET-II を用いた JIS 第一水準手書き漢字の認識, 信学技法, PRU93-151, pp.65-72 (1994).
- 4) 王, 荒井, 奥田: 増殖淘汰法および分類ネットを用いた手書き漢字認識, 情報処理学会論文誌, Vol.35, No.2, pp.272-280 (1994).
- 5) Saruta Kato Abe Nemoto: High Accuracy Recognition of ETL9B Using Exclusive Learning Neural Network-II (ELNET-II), Trans. IEICE, Vol.E79-D, No.5, pp.516-522 (1996).
- 6) 伊藤, 遠藤, 堀, 島村: 階層的印刷漢字認識システムにおける字種を複数クラスに登録する辞書構成法, 電子情報通信学会論文誌 (D-II), Vol.J78-D-II, No.6, pp.896-905 (1995).
- 7) 井藤, 大橋, 江島: 手書き文字認識における複数特徴を統合する認識器 EID3 の提案, 情報処理学会論文誌, Vol.37, No.4, pp.483-489 (1996).
- 8) 篠沢佳久, 大駒誠一: 二分木を利用した大分類・詳細認識型文字認識ニューラルネットワークの作成, 第 52 回情報処理学会全国大会論文集, Vol.2-233 (1996).
- 9) 篠沢佳久, 大駒誠一: 手書き文字認識における大分類のための決定木生成法, 第 54 回情報処理学会全国大会論文集, Vol.2-203 (1997).
- 10) キンラン, J.R.: AI によるデータ解析, 凸版印刷 (1995).
- 11) コホネン, T.: 自己組織化と連想記憶, シュプリンガー・フェアラーク東京 (1993).
- 12) 中野 馨: 入門と実習 ニューロコンピュータ, 技術評論社 (1989).
- 13) 大沢 裕, 山川健一: 図面の認識と理解, 昭晃堂 (1989).
- 14) 森 健一: パターン認識, 電子情報通信学会, コロナ社 (1988).
- 15) 萩田紀博: 手書き文字認識に関する研究, 慶應義塾大学理工学研究科学学位請求論文 (1986).

(平成 8 年 12 月 19 日受付)

(平成 9 年 10 月 1 日採録)

**篠沢 佳久 (学生会員)**

1994年慶応義塾大学理工学部管理工学部卒業。1996年同大学大学院理工学研究科管理工学専攻前期博士課程修了。現在同大学大学院理工学研究科管理工学専攻後期博士課程に在学。文字認識、パターン認識の研究に従事。電子情報通信学会学生会員。

**大駒 誠一 (正会員)**

1936年生まれ。1959年慶応義塾大学工学部卒業。慶応義塾大学理工学部管理工学部教授。工学博士。アルゴリズム、プログラミング言語、日本語情報処理に興味を持つ。現在は、日本の揺籃期のコンピュータのソフトウェア的復刻に取り組む。著書、「FORTRAN77」(サイエンス社)、「COBOLの基礎と応用」(サイエンス社)など。日本ソフトウェア科学会、計量国語学会、日本独文学会、ACM各会員。