プロトタイプ理論に基づく極小事例ベースの構成

大杉仁隆[†,☆] 上原邦昭[†,††]

事例に基づく学習では，原則としてすべての事例を事例ベースに記憶するため，事例の記憶量と類似事例の検索にかかる計算コストが重要な問題となる．本稿では，プロトタイプ理論に基づいて必要となる事例のみを記憶して，極小の事例ベースを構成するノンインクリメンタルなアルゴリズムを提案する．本アルゴリズムは，まず事例集合を分析して典型的な特徴のみからなる仮想的な事例（プロトタイプ）を生成する．さらに，カテゴリへの分類が困難な事例のみ選択している．このため，本アルゴリズムは記憶する事例が非常に少ないにもかかわらず，すべての事例を記憶する場合と同等の分類精度が維持できる．さらに，事例ベースには例外的な事例が含まれず，ノイズに強固であるという特徴がある．

Constructing a Minimal Instance-base by Storing Prototypical Instances

YOSHITAKA OOSUGI[†,☆] and KUNIAKI UEHARA[†,††]

Instance-based learning has been successfully applied to a wide variety of learning tasks. However, this technique retains the entire training set in the instance-base, having as result large memory requirements and slow execution speed. In this paper, we will introduce a new algorithm that constructs a minimal instance-base by storing prototypical instances for each category. In addition, a small number of instances that appear difficult to be predicted in their categories are stored into the instance-base. Experimental results showed that this algorithm recorded lower storage requirements and higher classification accuracies than the nearest neighbor algorithm and its derivatives on several domains. Furthermore, this algorithm is tolerant of the noisy instances.

1. はじめに

事例ベース学習（Case-Based Learning: CBL またはInstance-Based Learning: IBL）[1]は，事例集合（事例ベース）から一般的概念記述を帰納することなく，事例ベースそのものによって概念を表現する学習手法である．したがって，事例を分類する際には，ルールや決定木のように弁別に有用な特徴に着目して分類するのではなく，事例ベースから類似している事例を検索し，類似事例の属するカテゴリに分類するようになっている．このため，CBL はルールや決定木に比べると，事例の分類にかかる計算コストが大きくなるという問題がある．また，最も単純な CBL はすべての事例を記憶するため，事例の記憶量が大きくなるという問題もある．このため，冗長な事例やノイズとなる事例を削除し，記憶すべき事例を選択的に減らすことは，重要かつ有益な技術であると考えられている．

一般に，事例の選択的な記憶に関する研究は，機械学習の分野で数多く研究されている．たとえば，すべての事例をそのまま記憶しておき，類似した事例を用いて事例を分類する Nearest Neighbor 法[6]☆☆の研究がある．これらの研究では，事例の入力順序によって選択される事例が変化するインクリメンタルな手法が用いられることが多い．このため，後で選択した事例がすでに選択されている事例をカバーしていることも起こりうる．つまり，結果的には分類に必要のない事例が選択されることが起こりうるために，極小の事例

† 神戸大学工学部情報知能工学科
Department of Computer and Systems Engineering, Faculty of Engineering, Kobe University
☆ 現在，富士通関西通信システム株式会社
Presently with Fujitsu Kansai Communication Systems Limited
†† 神戸大学都市安全研究センター
Research Center for Urban Safety and Security, Kobe University

☆☆ 事例を記憶する前に何らかの処理（たとえば，数値で表される特徴の正規化）を事例に施す点で，最も単純な CBL と区別する研究者もいるが，多くの研究者は両者を区別していない．

ベースが得られるとは限らないという問題があった．

以上の問題を解決するために，本稿では，プロトタイプ理論に基づいて分類に必要な事例のみを記憶し，極小の事例ベース☆を構成するアルゴリズムSABIを提案する．本アルゴリズムは，まず，できるだけカテゴリを大きくカバーするような仮想的な事例（プロトタイプに相当する事例）を生成する．さらに，この仮想的な事例ではカバーできない事例を，カテゴリ間の境界との距離が近いものから順序づけて選択する．このため，記憶する事例が非常に少ないにもかかわらず，すべての事例を記憶した場合と同等の分類精度を維持することができる．また，ノンインクリメンタルなアルゴリズムであるため，事例の入力順序に影響を受けずに極小の事例ベースが得られるという特徴がある．さらに，仮に訓練事例集合に例外的な事例が含まれていても，カテゴリ間の境界の内側に存在する事例のみを選択するため，例外的な事例を自動的に排除することができる．この結果，得られた極小の事例ベースはノイズの影響を受けにくいという特徴がある．

2. 基本的な考え方

記憶すべき事例の選択は，Roschら[15]のプロトタイプ理論に基づいている．プロトタイプ理論によれば，それぞれのカテゴリにはプロトタイプと呼ばれる典型的な事例が存在している．そして，この典型的な事例を用いれば，同じカテゴリに属する多くの事例を正分類できると考えられている．たとえば，脊椎動物に含まれるカテゴリ「鳥類」について考える．鳥類に属している「すずめ」は，「空を飛ぶ」，「羽毛を持つ」，「虫を食べる」などの他の多くの鳥が持つ典型的な特徴を持っているため，鳥類の典型的な事例だといえる．

一方，人間はまず一般概念を用いて推論を行い，一般概念に当てはまらなければ，過去に経験した特殊な事例を用いて推論するといわれている[14]．この考え方にプロトタイプ理論を導入すると，「一般概念」はプロトタイプと見なすことができる．また，「一般概念に当てはまらない事例」は，プロトタイプでは正分類できない事例ということになる．

以上の考え方に基づいて，SABIではプロトタイプとして典型的な特徴からなる仮想的な事例を新たに生成している．この仮想的な事例を平均事例と呼ぶ．次に，平均事例では正分類できない事例を記憶する必要があるが，これらの中には分類に悪影響を与える事例やノイズとなる事例（例外的な事例）が含まれている可能性がある．このため，本稿では新たに境界事例という概念を導入し，カテゴリ間の境界の近くに存在し，いずれのカテゴリに属するか弁別が困難な事例を残すようにしている．

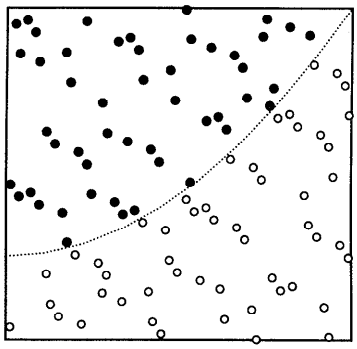
たとえば，鳥類に属している「ペンギン」である．「ペンギン」は，「空を飛ぶ」や「虫を食べる」といった特徴は持っておらず，「泳ぐ」や「魚を食べる」などの別のカテゴリである魚類に属する事例が持っているような特徴を持っている．このため，「ペンギン」は鳥類と魚類のどちらに属するか弁別が困難な事例であるといえる．このような事例を記憶しておくと，平均事例では正分類できない事例でも，同じカテゴリに属する類似事例が存在することになり，正分類が可能となる．このため，境界事例は非常に重要な事例だと考えられる．最終的に，平均事例と境界事例を記憶した事例ベースが極小の事例ベースとなる．

以上の考え方を，2次元の事例空間を用いて説明する．また，カテゴリは2種類あるものとする．図1(a)は，すべての事例を記憶している事例空間である．図中の点線は，カテゴリ間の境界を表しており，●と○は点線で分類されている事例を表すものとする．次に，平均事例のみを記憶した事例空間を図1(b)に，平均事例と境界事例を記憶した事例空間を図1(c)に示す．ここで，▲，△はそれぞれのカテゴリの平均事例を表しているものとする．また，実線は記憶している事例によって形成されるカテゴリ間の境界を表しているものとする．

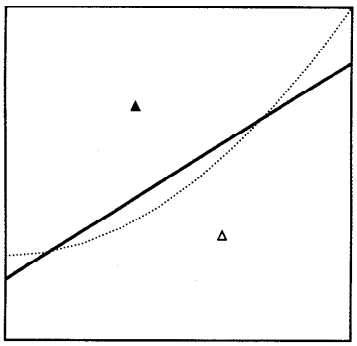
事例の分類は，実線で区切られた領域のどこに事例が存在するかによって行われるため，実線が点線に近似しているほど分類精度が高いことになる．図1(b)では，平均事例だけで多くの事例を正分類できるように見えるが，単純な境界しか形成されないため，点線付近に存在する事例は誤分類されることが分かる．これに対して，図1(c)では，平均事例に加えて境界事例を記憶しているため，形成されるカテゴリ間の境界を点線に十分近似できることが分かる．

これらのことから，SABIは，まず平均事例を記憶して目標となるカテゴリ間の境界に近似した境界を形成し，次に，少数の境界事例を用いて，形成された境界をさらに目標の境界に近似させるアルゴリズムだといえる．このため，分類に必要のない事例を記憶することなく，すべての事例を記憶する場合と同等の分類精度を維持することができる．さらには，例外的な事例が自動的に排除されるため，分類精度の向上が期待

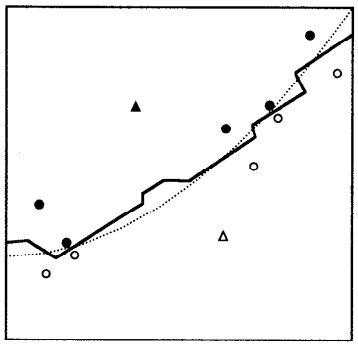
☆ 本来，各カテゴリに事例が1個のみの場合が極小の事例ベースとなるが，1個のみの事例ではうまく概念を表現できない場合が生じるため，できるだけ少ない例外的な事例も含めて構成されたものを極小の事例ベースと呼んでいる．



(a)



(b)



(c)

図1 SABIによる2事例の選択的な記憶
Fig. 1 Storing instances using SABI.

できる．

3. SABIアルゴリズム

3.1 平均事例

まず，本稿で用いる事例の記述形式を定義する．事例は，属性とその値からなる特徴とその事例が属するカテゴリからなる．たとえば，n 個の特徴を持つ事例 I は，以下のように表すものとする．

$$I = (c, a_1, a_2, \ldots, a_n) \quad (1)$$

ただし，c はカテゴリ，a_i は i 番目の属性の値を表している．

上原ら[19]の典型性に基づく概念学習アルゴリズム（Prototype-Based Learning Algorithm）では，Roschらの考え方を学習に応用し，特徴の出現頻度を典型性の度合として事例の分類を行っている．つまり，あるカテゴリにおいて，ある属性がとりうる値の中で出現頻度が最も高い値とその属性の組が，そのカテゴリの典型的な特徴だという考え方である．本稿でも同様に，カテゴリ c に属する事例の i 番目の属性がとりうる値のうち，出現頻度が最も高い値を c の平均事例の i 番目の属性値としている．また，属性値が数値の場合は，平均値を採用している．なお，平均事例 I_{ave} は以下のように表される．

$$\begin{aligned} I_{ave} &= (c, ave_1, ave_2, \ldots, ave_n) \\ ave_i &= \begin{cases} k_{ci} & \text{非数値属性} \\ \mu_{ci} & \text{数値属性} \end{cases} \end{aligned} \quad (2)$$

ここで，k_{ci} はカテゴリ c に属する事例の i 番目の属性がとりうる値のうち，出現頻度が最も高い値であり，μ_{ci} はカテゴリ c に属する事例の i 番目の値を平均した値である．

3.2 境界事例

一般に，ほとんどの問題領域ではカテゴリ間の境界が未知であり，境界を発見することは非常に困難である．本稿では，事例を1次元上に表現し，情報理論に基づいてカテゴリ間の境界を設定するようにしている．そして，設定した境界を基準に境界事例を選択するようにしている．

3.2.1 カテゴリと事例間の距離

Roschらは，事例と属するカテゴリの類似性に関する心理学実験を行い，同じカテゴリに属する事例と共通する特徴を多く持ち，別のカテゴリに属する事例と共通する特徴をあまり持っていない事例と属するカテゴリの類似性が高いという結果を導いている．つまり，事例と属するカテゴリの類似性は，同じカテゴリに属する事例との類似性だけでなく，他のカテゴリに属する事例との類似性も考慮した尺度として定義される．たとえば，ある事例と属するカテゴリ c の類似性は，c に属する事例の多くと類似しており，さらに c 以外のカテゴリ $\bar{c}$ に属する事例とほとんど類似していない場合に高くなる．このような考え方を用いると，同じカテゴリに属する事例とあまり類似していない事例の中でも，他のカテゴリに属する事例の多くと類似しているような境界事例を判別することができると考えられる．以上のことを考慮しつつ，事例と任意のカテゴリの類似性の度合である距離 Dis を定義する．

まず，事例と任意のカテゴリの単純な類似性を表す

類似度を定義する．この類似度は，事例の持つ各属性値がカテゴリを代表する属性値であることに対する妥当性を表す度合 *invalidity* の合計として定義している．*invalidity* は，値が小さいほど妥当性が高いことを表しているため，類似度は値が小さいほど類似性が高いことを表している．事例 I と任意のカテゴリ c の類似度 $dissimilarity(c, I)$ は，以下のように表される．

$$dissimilarity(c, I) = \sum_{i=1}^{n} invalidity(c, a_i)$$

$$invalidity(c, a_i) = \frac{\sum_{j=1}^{m} feature\_dissimilarity(b_{ij}, a_i)}{m} = \begin{cases} 1 - P_c(a_i) & \text{非数値属性} \\ \sigma_{ci}^2 + (a_i - \mu_{ci})^2 & \text{数値属性} \end{cases} \quad (3)$$

ここで，n は属性数，m はカテゴリ c の事例数を表している．$feature\_dissimilarity(b_{ij}, a_i)$ は，属性値が非数値属性ならば $b_{ij} = a_i$ のとき 0，$b_{ij} \neq a_i$ のとき 1，数値属性ならば b_{ij} と a_i の差の 2 乗と定義している☆．また，*invalidity* は c に属する各事例の i 番目の属性と a_i の非類似性 *feature_dissimilarity* の平均と定義している．このため，$invalidity(c, a_i)$ は非数値属性ならばカテゴリ c における属性値 a_i を持つ事例の出現頻度 $P_c(a_i)$ を用いて $1 - P_c(a_i)$ と簡略化できる．また，数値属性ならば i 番目の属性値の平均 μ_{ci} と標準偏差 σ_{ci} を用いて $\sigma_{ci}^2 + (a_i - \mu_{ci})^2$ と簡略化できる☆☆．

次に，I と c の類似度 $dissimilarity(c, I)$ と，I と c 以外のカテゴリ $\bar{c}$ の類似度 $dissimilarity(\bar{c}, I)$ を用いて，事例 I と任意のカテゴリ c の距離 *Dis* を定義する．事例があるカテゴリとあまり類似しておらず（$dissimilarity(c, I)$ が大），他のカテゴリと類似している（$dissimilarity(\bar{c}, I)$ が小）ほど $Dis(c, I)$ の値が大きくなるように，$dissimilarity(c, I)$ を分子，$dissimilarity(\bar{c}, I)$ を分母とした比で $Dis(c, I)$ を定義している．したがって，$Dis(c, I)$ は以下のように表される．

$$Dis(c, I) = \frac{dissimilarity(c, I)}{\min_{\bar{c}}[dissimilarity(\bar{c}, I)]} \quad (4)$$

ここで，分母は c を除くすべてのカテゴリの中で最小となる *dissimilarity* としている．

式 (4) を簡略化したものは Zhang の TIBL[17]における典型度でも用いられている．Zhang の典型度は，カテゴリが 2 種類の問題領域のみを対象とし，分子，分母はそれぞれ，カテゴリ c に属する事例との類似度の平均，他のカテゴリ $\bar{c}$ に属する事例との類似度の平均としている．また，概念形成システムの 1 つである Fisher の COBWEB[9]の概念階層の評価指標（Category Utility）にも同様の考え方が用いられている．すなわち，COBWEB では事例とカテゴリ c との類似性を，事例が c に属することが認められたときに，属性値 a_i を持っているという条件付き確率 $p(a_i|c)$ に反映されるものとしている．また，他のカテゴリ $\bar{c}$ との非類似性を，事例が属性値 a_i を持つことが認められたときに，c に属しているという条件付き確率 $p(c|a_i)$ に反映されるものとしている．さらに，Category Utility にはこれらの条件付き確率の積が用いられている☆☆☆．この積は，カテゴリ c との類似性が高く，カテゴリ $\bar{c}$ との類似性が低い場合に大きくなる値である．このことから，COBWEB でも SABI と同様の考え方が用いられていることになる．

3.2.2 カテゴリ間の境界の設定

前項で定義した *Dis* を用いれば，n（事例の持つ特徴数）次元の事例空間で表現されている事例を 1 次元の *Dis* 軸上で表現できるようになる．さらに，2 つのカテゴリに属する事例を *Dis* 軸上に表現すれば，この軸上でカテゴリ間の境界となる点（値）を設定することができる．本稿では，相互情報量による数値属性値の区間分割[18]と同じ方法を用いてカテゴリ間の境界を設定する．

まず，カテゴリが 2 種類の場合の境界の設定について説明する．たとえば，属性数が 2 種類の問題領域が**図 2** (a) のような 2 次元の事例空間で表されているとする．ここで，●はカテゴリ c に，○はカテゴリ $\bar{c}$ に属している事例を表しているものとする．これらすべての事例に対して，どちらか一方のカテゴリとの *Dis* を計算すると，図 2 (b) のように 1 次元の *Dis* 軸上に事例を並べ換えることができる．この例では，c と事例間の *Dis* を計算して軸上に表現している．このため，*Dis* が小さいほど事例は c に近く，逆に大きく

☆ *invalidity* が 0 と 1 の間の値をとるように，数値属性の値は最小値が 0，最大値が 1 となるようにあらかじめ線形に正規化している．

☆☆ $invalidity(c, a_i) = ((b_{i1} - a_i)^2 + \cdots + (b_{im} - a_i)^2)/m = a_i^2 + (b_{i1}^2 + \cdots + b_{im}^2)/m - 2a_i(b_{i1} + \cdots + b_{im})/m = a_i^2 + \mu_{ci} + \sigma_{ci} - 2a_i\mu_{ci} = \sigma_{ci}^2 + (a_i - \mu_{ci})^2$.

☆☆☆ このほかに，問題領域全体における各属性値の出現頻度や，カテゴリ c の出現頻度などが用いられている．

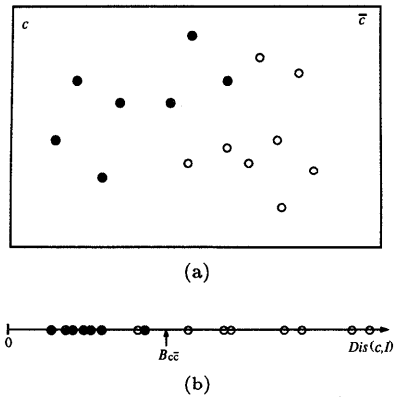


図2 2次元空間から1次元空間への変換
Fig. 2 Transformation from 2-dimensional space to 1-dimensional space.

なるほど $\bar{c}$ に近いことを示している．

このように1次元で表された事例空間では，カテゴリ間の境界を隣接する事例間に設定するようにする．すなわち，$(Dis(c,I_i)+Dis(c,I_{i+1}))/2$ を境界としている．ここで，i は Dis の大小でソートされた事例の番号を表しているものとする．しかし，境界として考えられる点は複数存在し，点によっては別のカテゴリ側に事例が多く含まれてしまう場合がある．このため，誤分類する事例数が減少すれば，境界の設定によって獲得される情報量，すなわち相互情報量が増加すると考え，これが最大となる点を境界としている．たとえば，図2(b)の例では，境界として考えられる14の点の中で相互情報量が最大となる点 $B_{c\bar{c}}$ に境界を設定するようにする．ただし，1次元上の事例集合を T とし，ある点で T_c と $T_{\bar{c}}$ に分割されるとすると，相互情報量 $M(T)$ は以下のように表される．

$$M(T)=I(T)-E(T)$$
$$I(T)=-P_T(c)\log_2 P_T(c)-P_T(\bar{c})\log_2 P_T(\bar{c})$$
$$E(T)=\frac{|T_c|}{|T|}I(T_c)+\frac{|T_{\bar{c}}|}{|T|}I(T_{\bar{c}}) \qquad (5)$$

ここで，$P_T(c)$ は T における c に属する事例の出現頻度を表しているものとする．

カテゴリが m 種類の場合には，カテゴリが2種類の場合の境界の設定方法を拡張し，隣接するカテゴリ間の境界をすべて求めて，互いを区別できるようにしている．たとえば，訓練事例中に m 種類のカテゴリ $c_1,c_2,\ldots,c_m$ があるとする．このとき，カテゴリ c_i に属する各事例に対して，それぞれ事例とカテゴリ間の類似性 *dissimilarity* が最小となる他のカテゴリを求める．ここで，カテゴリ c_j との *dissimilarity* が最小となった事例の集合を $S_{c_ic_j}$ と表した場合，カテゴリ c_i と c_j の境界は，$S_{c_ic_j}$ と $S_{c_jc_i}$ に含まれる事例間で，上記と同様の方法を用いて求めることができる．このようにして，すべてのカテゴリを互いに区別するためには，m 種類のうち2種類のカテゴリの組合せすべてに対して境界を設定しなければならないため，全部で ${}_mC_2$ 個の境界を求めなければならない．

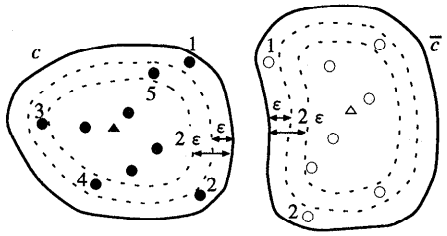


図3 境界事例が含まれる幅
Fig. 3 Interval domain that contains boundary instances.

3.2.3 境界事例の選択と最適な事例ベース

前項までの定義をふまえると，境界事例は Dis が設定される境界（値）に近い事例ということになる．SABIでは，設定される境界からある幅 ε の範囲内に含まれる事例を境界事例として選択するようにしている☆．したがって，境界事例が含まれる幅の大きさ ε は，構成される事例ベースの分類精度に大きく影響を及ぼす．そこで，本稿では境界事例が含まれる幅を ε ずつ増加させ，事例ベースとして複数の候補を構成するようにしている．

たとえば，**図3** のようなカテゴリが2種類の問題領域が存在するとする．ここで，●はカテゴリ c に，○はカテゴリ $\bar{c}$ に属している事例を表し，▲は c の，△は $\bar{c}$ の平均事例を表しているものとする．また，実線はカテゴリ間の境界を表しているものとする．まず，境界事例が含まれる幅をそれぞれ ε に設定すると，カテゴリ c に属する番号1，2の事例と c の平均事例，カテゴリ $\bar{c}$ に属する番号1，2の事例と $\bar{c}$ の平均事例からなる事例ベースの候補が構成される．このようにして構成される候補を，境界事例が含まれる幅が ε の何倍かを表す値 n を用いて，$Cd(n_c,n_{\bar{c}})$ と表すようにする．第1引数がカテゴリ c，第2引数がカテゴリ $\bar{c}$ の幅を表しているものとする．つまり，上記の例で構成される候補は，$Cd(1,1)$ と表すことができる．また，カテゴリ c の幅のみを倍の $2\times\varepsilon$ に設定すると，カテゴリ c に属する番号1から5の事例と c の平均事例，カテゴリ $\bar{c}$ に属する番号1，2の事例と $\bar{c}$

☆ 境界より内側に ε を設定しているため，境界の外側に存在する例外的な事例は自動的に排除される．

の平均事例からなる候補 $Cd(2,1)$ が構成される．なお，$Cd(0,0)$ は事例が1つも選択されず，平均事例のみからなる候補を表していることになる．

以上のようにして，カテゴリごとに n を増減させ，すべての組合せに対して $Cd(n_c, n_{\bar{c}})$ を構成している．そして，それぞれの $Cd(n_c, n_{\bar{c}})$ に対して，訓練事例☆の分類を行った際の誤分類率を求めて，最適な $Cd(n_c, n_{\bar{c}})$ を事例ベースとして選ぶようにしている．

次章以降の実験で用いたSABIでは，カテゴリ c の ε_c を $\sum_{i=1}^{m_c-1} |Dis(c,I_i) - Dis(c,I_{i+1})|/(m_c-1)$ に設定している．ここで，m_c は c に属する事例数であり，i はカテゴリごとに Dis の大小でソートされた事例の番号を表しているものとする．つまり，ε は隣合う事例間の距離の平均値である．この設定は，ε をなるべく小さい値にし，厳密に事例を選択して Cd を構成しようというヒューリスティックな考え方である．

しかし，本来ならば厳密に事例を選択することだけでなく，構成される Cd の数の増加にともなう計算コストの増大についても考慮しつつ ε を調節することが望ましい．このため，ε の設定に関してはさらに検討が必要だと思われる．最後に，本節までに述べたSABIアルゴリズムを図4に示す．

```
procedure SABI
input: TI: 訓練事例集合
output: IB: 事例ベース;
begin
  平均事例集合 AI ← ∅
  for カテゴリ c とカテゴリ c̄ の組
    c に属している c̄ に近い事例の集合 S_{cc̄} ← ∅
    c̄ に属している c に近い事例の集合 S_{c̄c} ← ∅
  TI をカテゴリごとの集合に分割する
  for 各カテゴリ c
    c の平均事例 I_ave を生成する
    AI ← AI ∪ {I_ave}
    for 事例 x ∈ カテゴリ c に属する事例
      類似度 dissimilarity が最小である c̄ を求める
      S_{cc̄} ← S_{cc̄} ∪ {x}
  for カテゴリ c とカテゴリ c̄ の組
    for 事例 x ∈ S_{cc̄} ∪ S_{c̄c}
      カテゴリ c との距離 Dis を求める
    境界 B_{cc̄} を設定する
  for カテゴリごとの n の組合せ (n: 正数)
    Cd(n_{c_1}, ..., n_{c_m}) ← AI ∪ ⋃_{i=1}^{m} BI(n_{c_i} × ε_{c_i})
      (BI(n × ε): n × ε の幅に含まれる境界事例集合)
    誤分類率 error(Cd(n_{c_1}, ..., n_{c_m})) を計算する
  error(Cd) が最小の Cd を IB として出力する
end
```

図4 SABIアルゴリズム
Fig. 4 SABI algorithm.

4. SABIの評価

4.1 他のアルゴリズムとの比較

本章では，さまざまな領域のデータベースに対してSABIを適用し，その評価を行う．1つ目の実験では，事例の選択を行ってもすべての事例を記憶する場合の分類精度を維持できるかどうかを検証するために，Nearest Neighbor 法との分類精度の比較を行っている．適用したデータベースは，UCI Machine Learning Repository[13]から得たものである．すべての結果は30回の試行の平均をとっている．1回の試行では，データベース中のすべての事例からランダムに5分の4を訓練事例とし，残りの5分の1をテスト事例として分類精度を求めている．

実験結果を**表1**に示す．表中のNN法は，Nearest Neighbor 法を表している．また，選択率とは訓練事例の中から選択される事例の割合を表しているものとする．したがって，選択率が N% とは，選択される事例の数が訓練事例数の N% であることを意味している．なお，Nearest Neighbor 法ではすべての訓練事例を選択しているため，選択率は100%ということになる．また，● 印は水準1%で平均の差の検定を行い，SABIの方がNearest Neighbor 法よりも分類精度が高いと検定されたものを示している．★ 印は，Nearest Neighbor 法の方がSABIよりも分類精度が高いと検定されたものを示している．

表1 Nearest Neighbor 法とSABIの分類結果
Table 1 Summary of experimental results for NN and SABI.

データベース名	NN法の分類精度	SABI	
		分類精度	選択率
breast-cancer	70.85%	★ 67.53%	28.70%
soybean	100.00%	★ 99.39%	27.78%
voting	87.90%	● 91.55%	29.79%
hayes-roth	69.18%	● 76.06%	1.85%
promoters	80.33%	77.95%	7.35%
tic-tac-toe	97.76%	★ 88.94%	9.66%
cleveland	77.45%	● 81.44%	28.76%
credit	80.19%	79.30%	14.61%
iris	94.89%	94.34%	30.00%
pima-diabetes	69.98%	71.26%	29.06%

表1の結果から，SABIは30%以下に事例を減らしても，ほとんどのデータベースにおいて分類精度の低下を3%程度までにおさえていることが分かる．voting，hayes-roth，clevelandでは分類精度の向上さえも見られる．これは，データベース中にノイズが含まれているため，Nearest Neighbor 法では残されてし

☆ 例外的な事例は，誤分類率を増加させる原因となるため，あらかじめ除いている．

表2 SABIとIB3の分類結果
Table 2 Summary of experimental results for SABI and IB3.

データベース名	SABI		IB3	
	分類精度	選択率	分類精度	選択率
breast-cancer	64.02%	20.01%	65.01%	20.66%
soybean	• 99.39%	• 0.00%	88.18%	27.88%
voting	91.67%	• 1.23%	91.02%	9.00%
hayes-roth	• 76.79%	• 0.00%	55.32%	9.65%
promoters	73.57%	• 2.00%	74.29%	20.17%
tic-tac-toe	81.42%	16.21%	80.20%	17.98%
cleveland	78.04%	• 11.85%	77.87%	13.19%
credit	78.91%	• 0.47%	80.63%	10.83%
iris	93.56%	• 4.94%	93.90%	11.51%
pima-diabetes	69.05%	15.69%	68.18%	16.32%

まうノイズを，SABIでは除去できていることを示している．

また，hayes-rothとpromotersでは，あまり事例が残されていないことが分かる．これは，カテゴリ間の境界が比較的単純であり，平均事例のみで十分に近似できるためと考えられる．一方，tic-tac-toeでは分類精度が10%程度低下している．これは，tic-tac-toeの問題領域が特徴の選言的標準形（Disjunctive Normal Form: DNF）☆を用いて表現できる概念であることが原因と考えられる[12]．つまり，特徴の出現頻度ではなく，特徴間の関係によってカテゴリが決定されているのである．このため，特徴の頻度情報を用いて事例の生成，選択を行っているSABIでは，逆に特徴間の関係が反映されないため，分類精度が悪くなってしまっていると考えられる．

2つ目の実験では，インクリメンタルなアルゴリズムと性能を比較し，SABIの有効性を検証する．ここでは，SABIと同様に分類に有効な事例を境界事例として選択を行うAhaのIB3[3),4)]と性能を比較する．IB3は，最初に入力される事例を記憶したのち，誤分類する事例を境界事例と見なして選択している．さらに，選択した事例の中から，頻繁に誤分類に使われる事例はノイズを含む事例として削除するようにしている．したがって，IB3にはプロトタイプの概念がなく，最初に入力される事例がカテゴリを代表する事例として働いている．このため，事例の入力順序によっては，例外的な事例が代表的な事例になる可能性があり，分類精度を大きく左右するおそれがある．

まず，SABIとIB3の分類精度の比較を行った．この結果，記憶されている事例にあまり共通性がないにもかかわらず，分類精度に明らかな差はほとんど見られなかった．そこで，次にSABIとIB3の分類精度にほとんど差がないようにして選択率を比較する実験を行った．この実験では，試行錯誤しながらSABIの選択率を変化させ，得られる分類精度がIB3とほぼ同一になった場合の選択率を比較している．適用したデータベース，実験方法などは，Nearest Neighbor法との比較実験と同じにしている．ここで，• 印は水準1%で平均の差の検定を行い，SABIの方がIB3より分類精度が高い，または選択率が小さいと検定されたものを示している．

表2から，ほとんどのデータベースにおいて分類精度をほぼ同一にした場合，SABIの選択率はIB3の選択率より下まわっていることが分かる．また，soybeanとhaycs-rothでは，選択率が0%，すなわち平均事例のみにもかかわらず，IB3よりも高い分類精度が得られている．このため，SABIはIB3で選択される無駄な事例をうまく排除し，必要な事例のみを適切に選択しているといえる．

4.2 計算コスト

前節の実験で示したように，他のアルゴリズムと比べると，分類精度や選択率といった観点から見た性能では，SABIの方に優位性が見られた．しかし，SABIはノンインクリメンタルなアルゴリズムであるため，インクリメンタルなIB3に比べると，事例ベース構成までにかかる計算コストが非常に大きくなるという問題がある．実際に，いくつかのデータベースに対する30回の試行が終了するまでに，IB3ではすべて数十秒で終了しているが，SABIでは数日かかることもあった☆☆．

以上の問題をO記法による計算コストを比較して検討する．まず，IB3では事例を1つ入力して事例ベー

☆ $(x_1 \wedge x_2) \vee (x_3 \wedge x_4) \vee (x_5 \wedge x_6)$ のように，and記述をor結合した記述形式．

☆☆ 利用した計算機は，SPARC Statoin 5〔CPU：microSPARCII（110 MHz），Memory：32 MB〕である．

スを更新するのに $O(|IB_i| \times |A|)$ の計算コストがかかるため，全体では $\sum_{i=1}^{|TI|} O(|IB_i| \times |A|)$ となる．なお，$|TI|$ は訓練事例数，$|IB_i|$ は i 番目の事例が入力された際にすでに事例ベースに記憶している事例数，そして $|A|$ は事例の属性数を表している．これに対して，SABI は j 番目の候補 Cd_j の誤分類率を計算するのに $O(|TI| \times |Cd_j| \times |A|)$ の計算コストがかかる．このため，全体では $\sum_{j=1}^{|Candidate|} O(|TI| \times |Cd_j| \times |A|)$ となる．なお，$|Candidate|$ は考えられる Cd の数であり，$|Cd_j|$ は Cd_j の事例数を表している．さらに，$|IB_i|$ と $|Cd_j|$ は $|TI|$ に比べると十分小さくとりうる値の範囲が比較的狭いため☆，定数と見なすと，全体の計算コストはそれぞれ $O(|TI| \times |A|)$，$O(|Candidate| \times |TI| \times |A|)$ と表せる．ここで，境界事例が含まれる幅の最大値が $n_{\max} \times \varepsilon$ のとき，$|Candidate|$ はカテゴリごとの $n_{\max}$ の積で表される．このため，$|Candidate|$ は $n_{\max}$ やカテゴリ数の増加にともない，非常に大きな値となる．この結果，SABI の方が IB3 よりも計算コストが非常に大きくなっていることが分かる．

5. 計算コストの削減

前章の実験では，SABI による事例の選択の有効性を示すことができた反面，事例ベース構成までにかかる計算コストが大きくなりすぎることが明らかとなった．この原因としては，3.2.3 項で述べたように ε を比較的小さい値に設定しているため，事例ベースの候補が非常に多くなってしまっていることと，これらの候補すべてに対して誤分類率を計算していることが考えられる．本章では，2 つ目の原因を解決して，計算コストを削減する手法を提案する．

SABI の事例ベースの構成は一種の探索問題であるため，効率的な探索手法を取り入れれば，結果として計算コストを削減できると考えられる．まず，事例ベースの候補を節点としたグラフを用いて，SABI の探索空間の例を示す．カテゴリが 2 種類 $(c, \bar{c})$ あり，ともに境界事例が含まれる幅の最大値が $3 \times \varepsilon$ である場合を考える．この問題領域における SABI の探索空間は，**図 5** の実線で表されるものとする．なお，点線部分の節点は 4.1 節の IB3 との比較実験で用いた SABI のように，選択率の小さい Cd から最適な事例ベースを得る場合に探索する必要のない節点とする．ただし，節点内に示されているのは，3.2.3 項の定義に基づいて候補を表現したものである．すなわち，$Cd(1,2)$ はカテゴリ c では ε_c，カテゴリ $\bar{c}$ では $2 \times \varepsilon_{\bar{c}}$ に境界事例が含まれる幅を設定して構成される候補を表している．

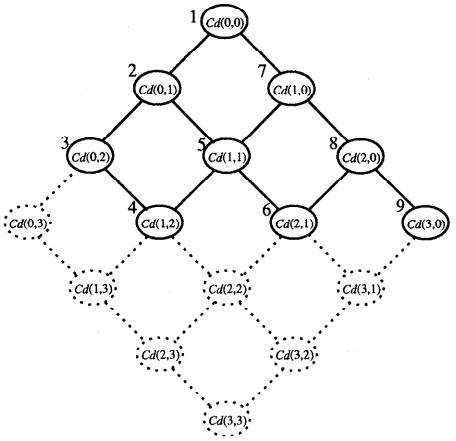


図 5 SABI の探索空間
Fig. 5 SABI's search space.

このように，SABI では考えられる節点をすべて探索している．しかし，すべての節点を探索する必要はなく，それ以上探索しても誤分類率の低下が見込めないと判断できる部分に関しては，探索を放棄するのが望ましい．そこで，バックトラックによって誤分類率の低下が見込めない節点の探索を放棄し，探索する節点を削減する 2 種類の探索手法を以下で提案する．

(1) Topdown バックトラック法：Topdown バックトラック法は，選択する事例を増やしながら，言い換えると，$n \times \varepsilon$ を大きくしながら探索を行う手法である．また，探索は節点 1 を始点とした深さ優先探索を行うものとする．このため，節点は番号順に探索されることになる．探索中，ある節点の誤分類率がその前に探索した節点の誤分類率よりも増加すれば，前の節点にバックトラックしてそれ以上深く探索しないようにする．これは，それ以上深く探索を行っても，ある特定のカテゴリの事例数が増加するのみで，カテゴリごとの事例数の偏りが大きくなってしまうためである．たとえば，図 5 の場合では，節点 7 の次に節点 8 を探索して誤分類率が増加した場合，次の節点 9 は探索せず，節点 7 にバックトラックするようにしている．

(2) Bottomup バックトラック法：Bottomup バックトラック法は，Topdown バックトラック法とは逆に，すべての事例集合から選択する事例を減らしながら探索を行う手法である．探索は，どのカテゴリの $n \times \varepsilon$ を大きくしても探索する必要のない節点に進んでしまうような節点を始点として，深さ優先探索を行

☆ SABI の $|Cd_j|$ については，IB3 との比較実験で行ったように，選択率が小さい Cd の集合から最適な事例ベースを得る場合を仮定している．

表3 SABIとT-SABI, B-SABIの分類結果
Table 3 Summary of experimental results for SABI, T-SABI and B-SABI.

データベース名	SABIの分類精度	T-SABI		B-SABI	
		分類精度	計算時間比	分類精度	計算時間比
breast-cancer	67.53%	⋆ 56.73%	• 0.38%	67.13%	• 8.86%
soybean	99.39%	99.09%	• 91.91%	99.29%	⋆ 105.93%
voting	91.55%	91.67%	• 0.46%	91.02%	• 30.97%
hayes-roth	76.06%	76.79%	• 0.64%	⋆ 61.37%	• 48.75%
promoters	77.95%	78.73%	• 2.07%	78.41%	• 39.22%
tic-tac-toe	88.94%	⋆ 66.52%	• 0.02%	89.53%	• 3.98%
cleveland	81.44%	78.20%	• 0.33%	81.44%	• 33.42%
credit	79.30%	78.91%	• 0.04%	78.09%	• 11.87%
iris	94.34%	94.44%	• 65.93%	94.33%	99.32%
pima-diabetes	71.26%	68.90%	• 0.02%	71.21%	• 18.39%

うものとする．開始点は1つとは限らないため，すべての開始点から探索をする必要がある．たとえば，図5の場合では，節点4，6，9を出発点として探索が行われる．このアルゴリズムもTopdownバックトラック法と同様の理由から，深く探索を進めていく際に，ある節点の誤分類率がその前に探索した節点の誤分類率よりも増加すれば，前の節点にバックトラックしてそれ以上深く探索しないようにしている．

上記の2つの探索手法を用いたSABIを，それぞれT-SABI，B-SABIと呼び，SABIとの分類精度と最適な事例ベース構成までにかかる計算時間の比較を行う．実験対象の問題領域は4章で用いたデータベースであり，実験方法なども同じにしている．結果を**表3**に示す．なお，それぞれの手法の計算時間はSABIの計算時間を100として換算した比で表している．また，• 印は，水準1%で平均の差の検定を行い，T-SABIとB-SABIの方がSABIよりも計算時間が小さいと検定されたものであり，⋆はSABIの方が分類精度が高い，または計算時間が小さいと検定されたものを示している．

表3から，T-SABIとB-SABIのどちらもsoybeanを除くほとんどのデータベースで計算時間を大幅に削減できている．また，分類精度も変化することなく，探索空間内で最適な事例ベースが見つかっている．soybeanで計算時間が削減できていないのは，事例がカテゴリごとにはっきりと分かれているため，ほとんどの節点において誤分類率が0%で等しかったことが原因と考えられる．このためT-SABI，またはB-SABIを用いても探索中にバックトラックが起こらず，探索する節点数を減らすことができていなかったと考えられる．また，3つのデータベースでT-SABI，またはB-SABIに適用した場合の方がSABIの分類精度よりも低くなっている．これは，評価値として用いた誤分類率が探索空間において多峰性を持っており，局所最適解が得られたものと思われる．つまり，最適解に至るまでの節点中に誤分類率が増加する節点があるため，最適解に到達する前にバックトラックしてしまっているのである．

結果的に，数日かかっていたSABIの計算時間をT-SABIでは数秒に，B-SABIでは数分に短縮することができた．また，T-SABIとB-SABIのどちらを用いても局所探索に陥るようなデータベースはなく，どちらかを用いれば最適解が得られている．このため，双方の探索手法をうまく組み合わせることができれば，局所探索を避けられる可能性がある．これらのことから，探索手法の導入による計算コストの軽減は有効であると思われる．

6. 関連研究

1章でも述べたように，事例の選択は機械学習の分野において多数研究されている．Biberman[5]とZhang[17]は，分類には典型的な事例が重要だとして，事例の典型度を用いた事例の選択手法を提案している．Bibermanの Prototypicality-Bascd Algorithmは，典型度の高い順に指定した事例数だけ選択する手法である．また，ZhangのTIBLは典型度が高い順に誤分類する事例を見つけ，この事例を正分類できる最も典型度の高い事例を選択する手法である．これらの手法では，すでに選択されている典型的な事例が，後で選択されるあまり典型的でない事例によってカバーされ分類に必要でなくなる場合がある．したがって，これらの手法では極小の事例ベースが得られない．このほかに，Skalak[16]はモンテカルロ法と山登り法を用いて事例の選択を行っており，分類精度を維持した非常に少ない事例の選択を実現している．この手法では，ランダムに構成する事例ベースの候補の数や選択する事例数が指定する数に固定されているため，指定する数によっては大幅に分類精度を下げる恐れがある．

また，Nearest Neighbor 法によって効率良く分類を行うために，パターン認識の分野においても古くから多数研究が行われている．Hart[11]が提案した CNN（Condensed Nearest Neighbor rule）は，種となる最初の訓練事例を人力後，誤分類される訓練事例を選択していく操作を，選択する事例がなくなるまで繰り返すというインクリメンタルな手法である．したがって，CNN では訓練事例をどのような順で分類していくかによって選択される事例が変化する．このため，選択される事例の集合が極小になるとは限らない．この欠点を解消するために，Gates[10]は CNN で選択した事例集合の中から，分類に不必要な事例を信頼度を用いて削減する RNN（Reduced Nearest Neighbor rule）を提案している．また，Dasarathy[7]のアルゴリズムはノンインクリメンタルであり，極小の事例集合 MCS（Minimal Consistent Set）を求めることができるようになっている．しかし，これらの手法では訓練事例に含まれるノイズとなる事例は考慮されておらず，ノイズに敏感な手法である．

7. おわりに

本稿では，生成した平均事例に加えて境界事例を選択的に記憶して極小の事例ベースを構成するアルゴリズム SABI を提案した．実験の結果，SABI はさまざまな問題領域で，すべての事例を記憶する場合と同等の分類精度が得られた．また，インクリメンタルなアルゴリズム IB3 との選択率の比較を行い，SABI が非常に少ない事例で高い分類精度を保つことができることを示した．さらに，T-SABI と B-SABI では，最適な事例ベースを探索する手法にバックトラック法を導入して，事例ベース構成までにかかる計算コストを大幅に軽減することができた．これらの手法では，探索空間が多峰性を持っている場合に探索が局所解に陥る可能性があるため，両手法を組み合わせるなど局所探索から抜け出すアルゴリズムの考案が必要である．

実験で多くのデータに適用した結果，SABI は特徴の頻度情報を用いて事例の生成，選択を行っているため，tic-tac-toe のように特徴間の関係が分類に影響する問題領域では分類精度を低下させてしまっていることが明らかとなった．対策としては，constructive induction[2]を導入して，頻度情報が有効となるように新しく特徴を構成することが考えられる．

さらに，SABI ではカテゴリごとに 1 つの平均事例を生成している．このため，カテゴリがさらにサブカテゴリに分けられるような領域では，分類精度を下げてしまう可能性がある．このため，クラスタリング手法を用いるなど，サブカテゴリがあるような領域への適用についての検討が必要である[8]．

最後に，SABI はノンインクリメンタルなアルゴリズムであるため，概念が不変の問題領域に対しては有効であると考えられる．しかし，問題領域によっては，時系列的な環境条件の変化によってカテゴリ間の境界が変化し，過去に有効だった事例が新しい状況のもとでは適さなくなる場合がある[20]．このような，環境条件の変化への対応も重要な課題となると考えられる．

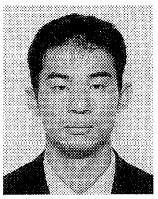
謝辞 本研究は，文部省科学研究費「発見科学」の援助を受けて行われたものである．本研究を進めるにあたり，数々の助言をくださいました神戸大学工学部情報知能工学科の森田浩助教授に感謝いたします．

参考文献

1) Aha, D.W.: Case-Based Learning Algorithms, *Proc. Case-Based Reasoning Workshop*, pp.147–158 (1991).
2) Aha, D.W.: Incremental Constructive Induction: Instance-Based Learning Algorithm, *Proc. 8th International Workshop on Machine Learning*, pp.117–121 (1991).
3) Aha, D.W. and Kibler, D.: Noise-Tolerant Instance-Based Learning Algorithms, *Proc. 11th International Joint Conference on Artificial Intelligence*, pp.794–799 (1989).
4) Aha, D.W., Kibler, D. and Albert, M.: Instance-Based Learning Algorithms, *Machine Learning*, Vol.6, pp.37–66 (1991).
5) Biberman, Y.: The Role of Prototypicality in Exemplar-Based Learning, Lecture Notes in Artificial Intelligence, Vol.912, pp.77–91, Springer-Verlag (1995).
6) Cover, T.M. and Hart, P.E.: Nearest Neighbor Pattern Classification, *IEEE Trans. Information Theory*, Vol.IT-13, pp.21–27 (1967).
7) Dasarathy, B.V.: Minimal Consistent Set (MCS) Identification for Optimal Neighbor Decision Systems Design, *IEEE Trans. Systems, Man, and Cybernetics*, Vol.SMC-24, No.3, pp.511–517 (1994).
8) Datta, P. and Kibler, D.: Symbolic Nearest Mean Classifiers, *Proc. AAAI-97*, pp.82–87 (1997).
9) Fisher, D.H.: Knowledge Acquisition via Incremental Conceptual Clustering, *Machine Learning*, Vol.2, pp.139–172 (1987).
10) Gates, G.W.: The Reduced Nearest Neighbor Rule, *IEEE Trans. Information Theory*, Vol.IT-18, No.3, pp.431–433 (1972).
11) Hart, P.E.: Condensed Nearest Neighbor Rule,

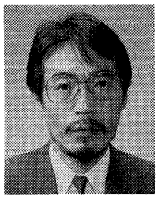
IEEE Trans. Information Theory, Vol.IT-14, No.3, pp.515–516 (1968).
12) Matheus, C.J. and Rendell, L.A.: Constructive Induction on Decision Trees, *Proc. 11th International Joint Conference on Artificial Intelligence*, pp.645–650 (1989).
13) Murphy, P.M. and Aha, D.W.: UCI Repository of Machine Learning Databases, Technical Report, University of California, Department of Information and Computer Science, Irvine, CA (1994).
14) Riesbeck, C.K. and Schank, R.C.: *Inside Case-Based Reasoning*, Hillsdale (1989).
15) Rosch, E. and Mervis, C.B.: Family Resemblances: Studies in the Internal Structure of Categories, *Cognitive Psychology*, Vol.7, pp.573–605 (1975).
16) Skalak, D.: Prototype and Feature Selection by Sampling and Random Mutation Hill Climbing Algorithms, *Proc. 11th International Machine Learning Conference*, pp.293–301 (1994).
17) Zhang, J.: Selecting Typical Instances in Instance-Based Learning, *Proc. 9th International Conference on Machine Learning*, pp.470–479 (1992).
18) キンラン, J.R.（著），古川康一（訳）：AIによるデータ解析，トッパン (1995).
19) 上原邦昭，谷澤正幸，前川禎男：典型性に基づく概念学習アルゴリズム，情報処理学会論文誌，Vol.35, No.10, pp.1988–1997 (1994).
20) 渡辺博芳，奥田健三：記憶量の制限による事例の忘却，人工知能学会誌，Vol.12, No.1, pp.144–151 (1997).

(平成10年2月27日受付)
(平成10年9月 7 日採録)



大杉 仁隆（学生会員）

昭和49年生．平成8年神戸大学工学部情報知能工学科卒業．平成10年同大学院自然科学研究科情報知能工学専攻博士前期課程修了．同年富士通関西通信システム（株）に入社．在学中は主に機械学習の研究に従事．



上原 邦昭（正会員）

昭和29年生．昭和53年大阪大学基礎工学部情報工学科卒業．昭和58年同大学院博士後期課程単位取得退学．同大学産業科学研究所助手，講師，神戸大学工学部情報知能工学科助教授を経て，現在，同大学都市安全研究センター教授．同大学情報知能工学科を兼担．平成元年より2年までOregon State University, Visiting Assistant Professor．平成6年より8年まで神戸大学総合情報処理センター副センター長．工学博士．人工知能，特に機械学習，マルチメディアデータベース，自然言語によるヒューマンインタフェースの研究に従事．1990年度人工知能学会研究奨励賞受賞．人工知能学会，電子情報通信学会，計量国語学会，日本ソフトウェア科学会各会員．