原著論文

知識表現を変化させる学習について†

松本 裕二*・馬野 元秀*・藤丸 雅弘*・瀬田 和久*

人間の学習過程では、始めは与えられたデータをそのまま覚えておき、それを用いて推論を行うが、与えられたデータが増えるにつれて、それらをまとめた知識を持つようになる。つまり、データ数が少ないときに適した知識の表現方法と多いときに適した表現方法は異なっており、データに応じて適切な知識を持つようになると考えられる。このような学習を表現するために、本論文では抽象度が異なるいくつかの知識表現法を用いて、これらを変化させながら推論と学習を進めていく方法を提案する。具体的にはデータがひとつずつ与えられるごとに現在の知識を用いて推論を行い、推論結果と正解を比べて不正解が連続した場合に、推論法の切り換えや知識の再構成や知識表現の切り換えを行うことにより学習を行う。これによって、状況にできるだけ適した知識表現と知識集合を用いるようにする。UCIの機械学習用データセットの手書き数字識別データを使用してシミュレーションを行った。その結果、知識の量を抑えながらある程度の推論性能を保つとともに、人間のように知識表現が切り換わっていく学習が行われることが分かった。

キーワード：知識表現の切り換え、逐次学習、知識獲得、推論、学習プラトール

1. はじめに

機械学習では一度に与えられたデータ集合に対してアルゴリズムを適用して知識を生成するのが普通であるが、人間の学習ではデータを少しずつ受け取りながら知識をつくりあげていく。

例えば、人間が学習を行う過程を考えると、学習の始めの段階では問題に対する知識を全く持っていないので、与えられたデータをそのまま覚えておく。そして、推論時にはそれまでに与えられたデータからよく似ているデータの答の部分の推論結果にすると考えられる。データの数が多くなると、すべてのデータをそのまま覚えられなくなり、整理された知識をつくり、その知識に基づいて推論を行うようになると考えられる。つまり、データ数が少ないときに適した知識の表現方法と多いときに適した表現方法は異なっており、データが増えるにつれて人間の知識の表現は具体的なものから抽象的なものに変化していくと考えられる。これを模式化したのが図1であり、始めの段階では与えられたデータをそのまま覚え、学習が進むにつれてif-thenルールや決定木などの形で知識を持つように切り換わっていく様子を示している。また、知識の表現形式を変えずに、与えられたデータを元にして知識を

図1 人間の学習の流れ

再構成したり、知識を変えずに推論の方法だけを変更することもあると考えられる。

そこで、本論文ではデータがひとつずつ与えられるごとに推論を行いながら学習も行う状況を考え、抽象度が異なるいくつかのルール形式の知識表現を用いて、推論法の切り換え、知識の再構成、知識表現の切り換えを行うことにより、データの量に応じて、知識の量を抑えながらある程度の推論性能を保つような推論と学習を進めていく方法を提案し、その有効性を示すことを目的とする。そのため、最終的に得られた知識だけを評価するのではなく、推論過程全体を通しての知識を評価する。

そして、判別問題に対して提案法によるシミュレーションを行った結果、知識の量を抑えながらある程度の推論性能を保つとともに、人間のように知識表現が切り換わっていく学習が行われることが分かった。

2. 推論 / 学習システム

本節では一般的な形式で推論法の切り換え、知識の再構成、知識表現の切り換えを行いながら推論と学習を進めていくシステムの枠組みについて述べる。

† Learning by Switching Knowledge Representations
Yuji MATSUMOTO, Motohide UMANO, Masahiro TOMARU
and Kazuhisa SETA

* 大阪府立大学大学院理学系研究科 情報数理学専攻
Department of Mathematics and Information Sciences, Graduate School of Science, Osaka Prefecture University

2.1 学習の概要

機械学習で使用される知識の表現形式にはさまざまなものが存在し、それぞれの表現形式に応じた生成法と推論法がある。本システムではある表現形式での知識の集合と生成法と複数の推論法を組み合わせたものを知識表現と呼ぶことにする。システムにはこのような知識表現が複数存在しており、推論/学習時にはこのうちのひとつを用いる(複数を用いる方法も考えられる)。また、与えられた入力データとその正解を蓄えておく領域(保存データ集合)を持つ。本システムの構成は図2のようになる。

図2 学習システムの構成

本システムの推論/学習は次のような手順で行なわれる。

1. **推論** 与えられたひとつの入力データに対して、現在使っている知識表現の推論法と知識集合を用いて、答を推論する。
2. **学習** 推論結果と与えられた正解とを比較し、必要に応じて推論法の切り換え、知識集合の再構成、知識表現の切り換えを行う。また、与えられた入力データと正解の組を保存データ集合に追加する。

保存データ集合において、保存するデータの量と質(量が有限のときにどのようなデータを優先して蓄えておくか)をどのように管理するかは興味深い問題であり、現在検討中である。

2.2 学習時の操作

同じ知識表現で推論を続けていると、その知識集合と推論法では誤って推論してしまうデータが連続して出現することがある。このような状況は特に知識の量

が少ない学習の初期段階や、学習対象のデータの傾向が変わったときによく起こる。そこで、これ以降の推論の結果をよくするために、本システムでは、(a)推論法の切り換え、(b)知識集合の再構成、(c)知識表現の切り換え、の3つの操作を行う。

これらは毎回実行されるわけではなく、指定された条件が満たされたときにのみ実行される。本論文では、この条件として不正解が連続した回数を用いる。すなわち、 m_1 回連続不正解のときに推論法の切り換えを、 m_2 回連続不正解のときに知識集合の再構成を、 m_3 回連続不正解のときに知識表現の切り換えを行う。一般的には、 $0 < m_1 \leq m_2 \leq m_3$ であると考えられる。この3つの操作を具体的に説明する。

(a) 推論法の切り換え

現在の知識表現の推論法の中から、妥当であると思われる推論法に切り換える。すなわち、現在の知識表現が持つすべての推論法を用いて、保存データ集合に対して推論を行い、その中で正解率が最大となる推論法に切り換える。

(b) 知識集合の再構成

現在の知識表現が持つ知識生成法を用いて、保存データ集合から知識集合をつくり直す。そして、つくられた知識集合が適切であったかどうかを評価値で判断する。再構成後の評価値が増加している(等しい場合を含む)ときは再構成後の知識で推論/学習を続ける。逆に、評価値が増加していないときは続けて知識表現の切り換えを行う。人間はデータ数が増えたときは、知識表現の抽象度を高くして、知識集合全体の量を減らしながらも一定の正解率を保とうとすると考えられるため、ある知識表現 a の評価値 E_a には正解率 c_a (保存データ集合に対して知識表現 a の知識集合を用いて推論したときの正解率) と知識集合の大きさ r_a による次の式(1)を用いる。

$$E_a = \alpha c_a + (1 - \alpha) \left(1 - \frac{r_a}{R} \right) \quad (1)$$

ここで、 R はその時点での保存データの数を表わす。 $(1 - \frac{r_a}{R})$ は知識集合の量(少なさ)を示しており、保存データの数だけ知識が作成されたときに0となる。正解率 c_a と知識集合の量 $(1 - \frac{r_a}{R})$ は区間 $[0, 1]$ 内の値となり、これらに対して α と $(1 - \alpha)$ で重み付けている。

(c) 知識表現の切り換え

まず、知識表現が持つ知識生成法を用いて、保存データ集合から知識集合を作成する。それとデフォルトの推論法(一般的にはもっとも簡単な推論法)を用いて、保存データ集合に対して推論を行い、正解率を求める。これをすべての知識表現に対して行い、式(1)

による評価値が最大となる知識表現に切り換える。評価値が最大のものが複数存在する場合は先に登録した知識表現(一般的にはもっとも簡単なもの)を選択する。

本手法は人間のような知識の表現方法が変化するような学習を実現するために、推論結果が不正解になることにより、推論法の切り換え、知識の再構成、知識表現法の切り換えを行う学習法である。データの量と内容に応じて知識表現を切り換えることにより、データ数が少ないときにも多いときにも知識の量を抑えながらある程度の推論性能を保つような学習を行うことができる。また、保存データ集合に最近のデータを中心に記憶しておくことで、時間的にデータの傾向が変化するようなデータにも対応することができる。

D. Normanは文献[1]において、D. Lumelhartとともに、人間の学習を付着(accretion)と構造化(structuring)と同調(tuning)の3つに分けており、付着は既存の記憶への知識の追加、構造化は記憶構造の形成、同調は具体的事例への知識の調整を行うとしている。本手法での知識集合の再構成を付着、知識表現の切り換えを構造化であると考えることができる。

複数の学習法を用いる手法としては、アンサンブル学習がある[2][3][4][5]。アンサンブル学習はPAC(Probably Approximately Correct)学習モデル[6]の流れを組んでおり、複数の学習器を用いて個々に学習させ、これらの推論を組み合わせることによってより汎化能力の向上した学習器を生成する手法である。代表的手法であるAdaBoost[2][3]は推論が間違えやすいところを重点的にリサンプリングし、そのデータ集合によって学習器を生成していく手法である。アンサンブル学習には他にもBagging[4]、Arcing[5]などがある。しかし、アンサンブル学習ではひとつの知識表現による複数の学習器(本手法での知識集合)を複数作成し、これらを組み合わせて推論を行っていくものであり、本手法のように与えられたデータが増えるにつれて複数の知識表現を切り換えて適切なものを用いる方法とは異なっている。本手法の推論法に、アンサンブル学習のように複数の知識を組み合わせる推論結果を生成するような拡張も考えられる。このとき、使用する知識がどのように変化していくかを調べることは興味深いと思われる。

3. 知識の表現の形式

本論文で考える判別問題は n 個の実数値属性に対してクラスを判別するものである。 i 番目の学習データ $x_i = (x_{i1}, x_{i2}, \dots, x_{in})$ に対してクラス c_i を決める。ここで、 $x_{i1}, x_{i2}, \dots, x_{in}$ はそれぞれ属性 $A_1, A_2, \dots,$

A_n の属性値であり、実数値をとる。クラス属性 C はラベル値 $C_1, C_2, \dots, C_m$ からなる集合であるとする。

判別問題を解くための知識を表現する方法は数多く存在し、それらの抽象度も異なっている。例えば、MBR(Memory Based Reasoning)では単純にデータをそのまま保持する[7]、if-thenルール[8]や決定木[9]はより抽象的に知識を表現しているといえる。フレーム[10]では複雑な構造を持つ知識を表現することができ、ニューラルネットワーク[11]も明示的な形式ではないが知識を表現していると考えられる。本論文では得られた知識集合が適切かどうかを人間が判断しやすいように、ルールとみなすことができる抽象度の異なる次の4つの知識表現を用いる。

- ・学習データと同じ形式
- ・ファジィ if-then ルール
- ・決定木
- ・ファジィ決定木

これら4つの知識表現について具体的に説明する。

(1) 学習データと同じ形式

この表現法では学習データをそのままルールとみなす方法(生成法 G_{11})とクラスタリングを行って、クラスタの代表点を用いる方法(G_{12})の2つがある。前者は人間がデータをそのまま記憶するという状況に対応しており、後者はそのデータをまとめているが、同じ表現形式を用いている。推論には推論データに似ている知識データを用い、もっとも近い知識データを用いる方法(推論法 R_{11})と、近傍のすべての知識データを用いる方法(R_{12})の2つがある。

ルール生成法 G_{11} : 単純追加

保存データ集合をそのままルール集合とする。このため、保存しているデータに対しては必ず正解となる。この生成法はデータ更新が簡単であるが、必要な記憶領域はデータ量に比例するので大きなデータ集合を扱うのは困難となる。

ルール生成法 G_{12} : クラスタリング

保存データ集合を各クラスごとにクラス以外の属性でクラスタリングを行い、各クラスタの代表となる点をルールとする。ここでは、階層的クラスタリング[12]を用いる。与えられたデータ1つずつを1クラスタとみなし、それを併合していくことによりクラスタリングを行う。このとき、距離によるしきい値を設けて、クラスタ同士の距離がしきい値よりも大きくなったときに処理を終了する。

データ $x = (x_1, x_2, \dots, x_n)$ と $x' = (x'_1, x'_2, \dots, x'_n)$ との距離 $d(x, x')$ には次の式(2)を用いる。

$$d(x, x') = \sqrt{\sum_{j=1}^n \{d_j(x_j, x'_j)\}^2} \quad (2)$$

ここで、 $d_j(x_j, x'_j)$ は属性 A_j の属性値 x_j と x'_j との距離であり、これまでに与えられたデータの属性 A_j での最小値 α_j と最大値 γ_j によって正規化した次の式(3)を用いて計算する。

$$d_j(x_j, x'_j) = \frac{|x_j - x'_j|}{\gamma_j - \alpha_j} \quad (3)$$

クラス $B_i (i=1, 2, \dots, M)$ を代表する点 x_i は、このクラスに含まれるデータの属性 A_j での平均値 β_{ij} を用いて、

$$x_i = (\beta_{i1}, \beta_{i2}, \dots, \beta_{in}) \quad (4)$$

とする。そして、 x_i とクラス C_i の組をルール集合とする。

この生成法の例を図3に示す。2属性3クラスの18個のデータに対して各クラスごとにクラスタリングを行っており、同じクラスに属するデータを点線で囲っている。4つのクラスから代表点 x_1, x_2, x_3, x_4 を作成し、 $\{(x_1, C_1), (x_2, C_2), (x_3, C_2), (x_4, C_3)\}$ をルール集合とする。

図3 クラスタリング

推論法 R_{11} : 1-近傍法

入力データ x に対して、式(2)による距離が最小となるルールをひとつ選択し、その結論部(クラス)を推論結果とする。複数存在する場合には、データとして

与えられた時期が古いものを優先する。

推論法 R_{12} : 近傍データの重み付けによる方法

ある距離 a 内に含まれるデータから総合的に判断する。入力データ x に対するルール x'_i のクラス C_i の確信度 g_i は次の式(5)から求め、その合計が最大となるクラスを推論結果とする。

$$g_i = \max(a - d(x, x'_i), 0) \quad (5)$$

この推論法の例を図4に示す。入力データ x に対してもっとも近いルールはクラス C_3 のものであるが、入力データとの距離が近いルールの多いクラス C_2 が推論結果となる。範囲内にルール点がひとつも存在しない場合には推論結果として「不明」を返す。

図4 近傍データの重み付けによる推論

(2) ファジィ if-then ルール

このルール表現にはファジィ if-then ルール[8]を用いる。結論部は確信度の付いたクラス属性の集合とし、 i 番目のルールは次のように表すことができる。

$$r_i: \text{if } A_1 \text{ is } F_{i1} \ \& \ A_2 \text{ is } F_{i2} \ \& \ \dots \ \& \ A_n \text{ is } F_{in} \\ \text{then } C_1: w_{i1}, C_2: w_{i2}, \dots, C_m: w_{im}$$

この知識表現の生成法には、属性ごとに決められた数で分割してファジィ集合を作成し、それらを組み合わせる方法(G_{21})と、全属性クラスタリングを行って得られたクラスごとに if-then ルールを生成する方法(G_{22})の2つがある。推論法には簡略型ファジィ推論を用い、 $\min\text{-}\min\text{-}\max(R_{21})$ と $\times\text{-}\times\text{-}\times\text{-}\times(R_{21})$ の2つがある。

ルール生成法 G_{21} : 属性ごとの分割

保存データ集合から、属性 $A_j (j=1, 2, \dots, n)$ ごとに

M 個($M \geq 2$)のファジィ集合を作成し、それを組み合わせて M^n 個のルールを作成する。

保存データ集合の属性 A_j での最小値 α_j と最大値 γ_j より、区間 $[\alpha_j, \gamma_j]$ を $M+1$ 等分する点 $p_{jk}(k=1, 2, \dots, M)$ を用いて次の式(6)をメンバーシップ関数として持つファジィ集合 F_{jk} を作成する。

$$\mu_{F_{jk}}(x_j) = \begin{cases} Z(x_j; p_{j1}, p_{j2}) & \text{if } k=1 \\ \Pi(x_j; p_{j,k-1}, p_{jk}, p_{j,k+1}) & \text{if } k=2, \dots, M-1 \\ S(x_j; p_{j,k-1}, p_{jk}) & \text{if } k=M \end{cases} \quad (6)$$

ここで、関数 Z , Π , S は次の式(7)から式(9)に示す三角型のメンバーシップ関数である。

$$Z(x; p_1, p_2) = \begin{cases} 1 & \text{if } x < p_1 \\ \frac{p_2 - x}{p_2 - p_1} & \text{if } p_1 \leq x < p_2 \\ 0 & \text{if } p_2 \leq x \end{cases} \quad (7)$$

$$\Pi(x; p_1, p_2, p_3) = \begin{cases} 0 & \text{if } x < p_1 \\ \frac{x - p_1}{p_2 - p_1} & \text{if } p_1 \leq x < p_2 \\ \frac{p_3 - x}{p_3 - p_2} & \text{if } p_2 \leq x < p_3 \\ 0 & \text{if } p_3 \leq x \end{cases} \quad (8)$$

$$S(x; p_1, p_2) = \begin{cases} 0 & \text{if } x < p_1 \\ \frac{x - p_1}{p_2 - p_1} & \text{if } p_1 \leq x < p_2 \\ 1 & \text{if } p_2 \leq x \end{cases} \quad (9)$$

$M=3$ のときの例を図5に示す。また、属性 A_1 と属性 A_2 をもつデータから、各属性で3個のファジィ集合を用いる場合の例を図6に示す。

これらのファジィ集合を各属性ごとにすべて組み合わせて、ファジィif-thenルールを作成する。結論部には条件部のファジィ集合のメンバーシップ値がすべ

図5 三角型メンバーシップ関数 ($M=3$)

て0.5以上となる領域に存在するデータの、クラスごとの割合を用いる。この領域にデータがない場合には、ルールはつくらない。図6の F_{12} と F_{22} の組み合わせに対する例を図7に示す。 $F_{12}(x_1) \geq 0.5$ かつ $F_{22}(x_2) \geq 0.5$ となる領域には C_2 のデータが2個と C_3 のデータが1個存在するため $w_2=2/3$, $w_3=1/3$ となり、次のルールが得られる。

if A_1 is F_{12} & A_2 is F_{22}
then $C_2 : 0.67, C_3 : 0.33$

この手法で作成されたファジィ集合は、データ空間のすべてをカバーするとは限らず、推論法 R_{21} と R_{22} では範囲外のデータに対しては「不明」を返す。

図6 属性ごとのファジィ集合の作成

図7 ファジィ if-then ルールの結論部の計算

ルール生成法 G_{22} : クラスタリングによるファジィルール作成

保存データ集合をクラスごとに、生成法 G_{12} と同じ手法で階層的クラスタリングを行う。得られたクラス $B_1, B_2, \dots, B_M$ からファジィ集合を作成する。クラス B_i の属性 A_j での最小値 α_{ij} と最大値 γ_{ij} のときにメンバーシップ値が0.5となり、平均値 β_{ij} で1となるメンバーシップ関数を持つファジィ集合 F_{ij} を用いて、ルールを生成する。具体的には次の式(10)で定義する。

$$\mu_{F_{ij}}(x_j) = \Pi(x_j; 2\alpha_{ij} - \beta_{ij}, \beta_{ij}, 2\gamma_{ij} - \beta_{ij}) \quad (10)$$

ここで、 Π は式(8)で定義した関数である。

ファジィ集合 $F_{ij}(i=1, 2, \dots, M)$ からクラス B_i のルール r_i を作成する。

r_i : if A_1 is F_{i1} & A_2 is F_{i2} & A_n is F_{in}
then c_i

ここで、 c_i はクラス B_i のクラスである。データ数が1のクラスからはルールを作成しない。図8では、クラス B_1 からファジィ集合 F_{11} と F_{12} を作成してルール

if A_1 is F_{12} & A_2 is F_{22}
then C_1

を生成している。

この手法で作成されたファジィ集合は、データ空間のすべてをカバーするとは限らず、推論法 R_{21} と R_{22} では範囲外のデータに対しては「不明」を返す。

図8 クラスタからのファジィ集合の作成

推論法 R_{21} : min-min-max法

min-min-max法による簡略型ファジィ推論[13]を行い、メンバーシップ値が最大となるクラスを結論とする。

推論法 R_{22} : ×-×-十法

×-×-十法による簡略型ファジィ推論[13]を行い、メンバーシップ値が最大となるクラスを結論とする。

(3) 決定木

決定木はルールの条件部の共通するものをまとめて、順序をつけることにより木構造で表現したもので、葉でない節点(判別節点)は属性を持ち、その子節点への辺にはその属性の値を判別条件として持ち、葉節点には結論となるクラスを持つ。ルートから枝をたどることにより推論結果が得られる。

ルール生成法 G_{31} : ID3による決定木の作成

ID3のアルゴリズム[9]を用いて決定木を生成する。クラス占有率とデータ数によるしきい値を設け、生成中に節点におけるクラス占有率が一定の値以上となるか、データ数が一定数以下になる場合には、この節点を葉とし、占有率が最大のクラスを結論とする。

数値属性に対しては、はじめに属性による階層的クラスタリングを行ってクラスタを作成し、隣り合うクラスタの代表値(平均値)同士の間接点を分割点として判別条件をつくっておく。

推論法 R_{31} : 決定木による推論

ルート節点から節点の属性を調べて、その値と等しい辺をたどっていき、葉節点に達したときにその葉節点のクラスを推論結果とする。

(4) ファジィ決定木

ファジィ決定木は決定木をファジィ化したもので、判別節点から子節点への辺にはファジィ集合の条件を持ち、葉節点には結論として複数のクラスとその確信度を持つ。

ルール生成法 G_{41} : ファジィID3による決定木の作成

ファジィID3のアルゴリズム[14]を用いて決定木を生成する。数値属性に対してはその属性 A_j での階層的クラスタリングを行って M 個のクラスタを作り、それぞれのクラスタにおける代表値(平均値)をそれぞれ $p_{j1}, p_{j2}, \dots, p_{jM}$ ($p_{j1} \leq p_{j2} \leq \dots \leq p_{jM}$)として、式(6)をメンバーシップ関数として持つファジィ集合を用いる。

推論法 R_{41} : ファジィ決定木による推論

ファジィ決定木を使用した推論[14]を行う。これは×-+-十法による簡略型ファジィ推論と同様の推論で、ルート節点から判別条件が0より大きくなる節点をすべてたどっていく。到達したすべての葉節点において、結論部のクラスの確信度に、葉節点に至る各辺

でのメンバーシップ値の積を掛ける．これをクラスごとに合計して，これが最大のクラスを推論結果とする．

以上の4つの知識表現におけるルールの生成法と推論法をまとめると表1のようになる．本論文では表1のルール生成法と推論法を組み合わせた表2に示す6つの知識表現を用いる．

知識表現 K_1 は保存データすべてを蓄えるので，保存データに対しては必ず正解するが，ルール数は保存データに比例して大きくなってしまふ．知識表現 K_3 は得られるルール中のファジィ集合がラベル付けされているので分かりやすいが，属性数に応じてルール数が多くなってしまふ．それ以外の知識表現は保存データに対しては必ず正解するとは限らないが，知識表現 K_1 と K_3 に比べるとルール数はかなり少なく，平均的には緩やかに増加していく．

表1 知識の表現形式による生成法と推論法の組み合わせ

知識表現	生成法	推論法
学習データと同じ形式	G_{11} データ点をルールとする	R_{11} 最も類似しているルール
	G_{12} クラスタリングを行いその代表値をルールとする	R_{12} 類似度による重みを考慮する
ファジィルール	G_{21} 各属性ごとのクラスタリングによるファジィ集合を作成	R_{21} ファジィ推論(min-min-max法)
	G_{22} 金属性でのクラスタリングによるファジィ集合を作成	R_{22} ファジィ推論(X-X-+法)
決定木	G_{31} ID3による決定木	R_{31} 決定木による推論
ファジィ決定木	G_{41} ファジィID3による決定木	R_{41} ファジィ決定木による推論

表2 本システムで使用する知識表現

知識表現	ルール生成法	ルール推論法
K_1	G_{11}	R_{11}, R_{12}
K_2	G_{12}	R_{11}, R_{12}
K_3	G_{21}	R_{21}, R_{22}
K_4	G_{22}	R_{21}, R_{22}
K_5	G_{31}	R_{31}
K_6	G_{41}	R_{41}

4. シミュレーション

本手法の推論／学習の様子を見るために，シミュレーションを行った．学習データとして，UCI(University of California, Irvine)の機械学習用データセット[15]から手書き数字識別データを使用した．これは16属性(0から100までの整数値)10クラスであり，訓練データは7494個，検証データは3498個ある．今回は検証データの中から無作為にひとつずつデータを取り出してシステムに与えることを600回繰り返して，正解率とルール数の変化，推論法の切り換え，ルールの再構成，知識表現の切り換えがどのような状況で行われる

かを調べた．ここでの正解率には，学習の開始からその時点までの推論の正解の割合である累積正解率を用いる．推論法の切り換え，ルールの再構成，知識表現の切り換えを行う連続不正解数はそれぞれ $m_1=1$ ， $m_2=2$ ， $m_3=3$ とし，評価値の式(1)のパラメータは $\alpha=0.8$ とし，知識の大きさ r_a は学習データと同じ形式の知識表現ではデータの数(またはクラスタの数)とし，ファジィif-thenルールではルール数とし，決定木とファジィ決定木では葉節点の数とした．生成法 G_{21} における属性の分割数は $M=3$ とした．クラスタリングは重心法(しきい値は0.3)を用い，類似度の重みを考慮する推論法 R_{12} のパラメータは $a=0.2$ とした．決定木作成時の最小占有率は0.8，葉節点となる最小データ数は5とし，ファジィ決定木作成時の最小占有率は0.6とした．

与えるデータの順番によって結果が変わるため，異なるデータ順での4つのシミュレーション結果を図9の(a)から(d)に示す．実線は正解率で破線はルール数を示している．実線上の点は推論の切り換え，破線上の点はルールの再構成，グラフ上の垂直線は知識の切り換えが，その回の学習時に行われたことを示している．このとき，別の知識表現が使用された場合には，垂直線の右上に知識表現の切り換えで選択された知識表現を示している．なお，開始時は知識表現 K_1 である．

図9(a)の序盤を詳しく見ると，最初のデータに対しては「不明」を返して推論法の切り換えを行い，2個目のデータも「不明」でルールの再構成を行った．このとき，ルール数は2となり，評価値は $0.2(=0.8 \times 0 + 0.2 \times (1 - 0/2))$ から $0.8(=0.8 \times 1 + 0.2 \times (1 - 2/2))$ へ増加したのでそのまま学習を続けた．3個目のデータも不正解で知識表現の切り換えを行い，評価値はそれぞれ $E_1=0.8(=0.8 \times 1 + 0.2 \times (1 - 3/3))$ ， $E_2=0.8(=0.8 \times 1 + 0.2 \times (1 - 3/3))$ ， $E_3=0.8(=0.8 \times 1 + 0.2 \times (1 - 3/3))$ ， $E_4=0.2(=0.8 \times 0 + 0.2 \times (1 - 0/3))$ ， $E_5=0.4(=0.8 \times 0.333 + 0.2 \times (1 - 1/3))$ ， $E_6=0.4(=0.8 \times 0.333 + 0.2 \times (1 - 1/3))$ となり， K_1 が再び選択された．

4個目と6個目のデータで不正解となり推論法の切り換えを行うが R_{11} をそのまま使用した．7個目のデータも不正解でルールの再構成を行ってルール数は7となり，評価値は $0.571(=0.8 \times 0.571 + 0.2 \times (1 - 3/7))$ から $0.8(=0.8 \times 1 + 0.2 \times (1 - 7/7))$ へ増加したのでそのまま学習を続けた．9個目で不正解となり推論法の切り換えを行うが R_{11} をそのまま使用し，10個目のデータも不正解でルールの再構成を行ってルール数は7となり，評価値は $0.7(=0.8 \times 0.8 + 0.2 \times (1 - 7/10))$ から $0.8(=0.8 \times 1 + 0.2 \times (1 - 10/10))$ へ増加したのでその

(a)

(b)

(c)

(d)

図9 1回の学習での累積正解率とルール数の変化

まま学習を続けた。11個目のデータも不正解で知識表現の切り換えを行い、評価値はそれぞれ $E_1=0.8(=0.8 \times 1 + 0.2 \times (1-11/11))$, $E_2=0.818(=0.8 \times 1 + 0.2 \times (1-10/11))$, $E_3=0.8(=0.8 \times 1 + 0.2 \times (1-11/11))$, $E_4=0.327(=0.8 \times 0.182 + 0.2 \times (1-1/11))$, $E_5=0.582(=0.8 \times 0.545 + 0.2 \times (1-3/11))$, $E_6=0.655(=0.8 \times 0.636 + 0.2 \times (1-3/11))$ となり、 K_2 が選択された。

このように、不正解が連続した回数を用いて、人間の学習のように知識表現が切り換わっていくことがわかった。また、(a)や(c)のように学習初期の正解率は低下することで知識表現の切り換えが起こり、その後上昇する。これは学習プラトーと概念の再構造化(より適切な記憶構造による新しい学習)[1]に対応していると考えることができる。与えるデータの順序によっては、(b)のように知識表現 K_2 を使用しない場合や、(d)のように2つの知識表現の切り換えを繰り返す場合もある。

このシミュレーションでどの知識表現がどれくらいの割合で選択されていくかを見るために、先ほどと同じパラメータで、与えるデータの順番を変えて推論/学習を1000回行って、1推論ごとの知識表現の使用割合を求め、これを積み上げ折れ線グラフで表わしたものを図10に示す。左から K_1 , K_2 , K_4 , K_5 の順に知識表現が切り換わっている。なお、 K_3 と K_6 は初期にわずかに使用されている(例えば推論回数20のときに K_3 は7回、 K_6 は3回)。

学習初期には知識表現 K_1 が選ばれる傾向がある。これは評価式(1)において、知識表現 K_1 は正解率が必ず1となり、データ数が少ないのでルール数は他の知識表現と大差がなく、さらに評価値が同じ場合には K_1 を優先させているためである。しかし、知識表現 K_1 は評価値が常に α である(常に正解率が1でルール数が R となる)ため、徐々に他の知識表現の評価値が高くなる。

他の知識表現の中で、初期に正解率が高くなるのは知識表現 K_2 と K_3 である。ルール数については、知識表現 K_2 と K_4 がもっとも少なくなる。したがって、評価式(1)の $(1-\alpha)(1-\frac{r_a}{R})$ の部分の差が大きくなると、知識表現 K_2 が選ばれるようになる。

そこから、さらに学習を進むと知識表現 K_2 の正解率より知識表現 K_4 の正解率が高くなる。 K_2 と K_4 のルール数は同程度であるため知識表現 K_4 が選ばれやすくなる。 K_5 は K_4 よりも正解率が高いが、ルール数もやや多くなるので、評価式(1)の $(1-\alpha)(1-\frac{r_a}{R})$ の部分の影響が小さくなる後半に選ばれやすくなる。なお、知識表現 K_4 と K_5 は保存データ数が少ない学習

初期には正解率がかなり低くなってしまうため、選ばれることはない。

K_3 の生成法 G_{21} は従来よく用いられている属性ごとの分割によるルール作成であり、 M^n 個のルールを作成するため G_{11} 以外の生成法と較べるとルール数がかかなり大きくなってしまふ。そのため評価値が低くなり、選ばれなくなっている。 K_6 は K_5 と同程度の正解率だが、このパラメータ設定ではルール数が2倍から3倍になってしまうため選ばれない。

図10では、保存データ数が少ないときに正解率が高い知識表現 K_1 と K_2 を、保存データ数が多いとそれらよりも正解率が高くなる知識表現 K_4 と K_5 を用いるようになっていることが分かる。不正解が連続することで評価式(1)の値から、データ数が少ないときに適したものからデータ数が多いときに適したものへと知識表現が切り換わっていることが分かる。

また、手書き数字認識データに対して推論/学習を1000回行ったときの、推論回数ごとの式(1)による評価値の平均値、最大値、最小値の変化を図11に示す。平均値が単調に増加していることがわかる。

図10 知識表現の使用割合の変化

図11 評価値の平均の変化

5. おわりに

推論結果が不正解になることで、推論法の切り換え、知識の再構成、知識表現法の切り換えを行い、そのときの状況にできるだけ適した知識表現を用いることを目的とした学習法を提案した。提案手法によってシミュレーションを行い、正解率とルールの大きさを考慮しながら知識表現を切り換えることで、データ数が少ないときに適したものからデータ数が多いときに適したものへと知識表現が切り換わるようになり、知識の量を抑えながらある程度の推論性能を保つような推論/学習過程を得ることができた。

今後の展開については現在3つ考えている。1つめは保存データ数を制限して必要なデータのみを残していく場合についての実験と考察である。保存データ数を制限することは、人間の推論/学習過程のモデル化として自然であるだけでなく、最近のデータだけを記憶しておけば時間的に傾向が変化するデータにうまく対処できると考えられるし、不正解であったデータを中心に記憶しておけば無駄な記憶を減らすことができ、大規模なデータにも対応することができると考えられる。

2つめは学習法自体の改良である。例えば、ルールの生成法や推論法にはパラメータを必要とする場合がある。現在は、これらの値は学習開始前にユーザが決めたものを使用し、学習中に値の変更はしていない。これらの値も学習中に変更することで、効率のよい学習を行うことができると考えられる。

3つめは推論法の切り換え、知識の再構成、知識表現法の切り換えをどのようなときに行うのがよいかを学習することである。これは推論/学習過程の一連の試行からどの状況でどの操作を行えば推論/学習過程がよくなるかを学習する。この学習には明確な正解が存在しないため、これを実現するために強化学習などを用いることを考えている。

参考文献

- [1] Norman, D.: *Learning and Memory*, W. H. Freeman and Company, 1982. [邦訳] 富田達彦 訳: 認知心理学入門—学習と記憶—, 誠信書房, 1984.
- [2] Schapire, R. E.: A Brief Introduction to Boosting, *Proceedings of the 16th International Joint Conference on Artificial Intelligence*, pp.1401-1405, 1999.
- [3] Freund, Y. and Schapire, R. (安倍直樹 訳): A Short Introduction to Boosting, *人工知能学会誌*, Vol.14, No.5, pp.771-780, 1999.
- [4] Breiman, L.: Bagging Predictors, *Machine Learning*, Vol. 24, No. 2, pp.123-140, 1996.
- [5] Breiman, L.: Arcing the Edge, Technical Reports 460,

- Department of Statistics, University of California, Berkeley, 1997.
- [6] Valiant, L. G.: A Theory of the Learnable, *Communications of the ACM*, Vol. 27, No.11, pp.1134-1142, 1984.
 - [7] Waltz, D. and Kasif, S.: On Reasoning from Data, *ACM Computing Surveys*, Vol.27, No.3, pp.356-359, 1995.
 - [8] 水本雅晴: ファジィ推論法, システムと制御, Vol.28, No.7, pp.436-441, 1984.
 - [9] Quinlan, J. R.: Induction of Decision Trees, *Machine Learning*, Vol.1, No.1, pp.81-106, 1986.
 - [10] Minsky, M.: A Framework for Representing Knowledge, in Winston, P. H. (ed.): *The Psychology of Computer Vision*, McGraw-Hill, pp.211-277, 1975. [邦訳] 白井良明, 杉原厚吉 訳: コンピュータービジョンの心理, 産業図書, 1979.
 - [11] 今井兼憲: ニューラルネットワークコレクション, 共立出版, 1990.
 - [12] Anderberg, M. R.: *Cluster Analysis for Applications*, Academic Press, Inc., 1973. [邦訳] 西田英郎 監訳, 佐藤嗣二 他訳: クラスター分析とその応用, 内田老鶴園, 1988.
 - [13] 日本ファジィ学会編: ファジィ論理, 講座ファジィ 4, 日刊工業新聞社, pp.217-254, 1993.
 - [14] Umano, M., Okamoto, H., Hatono, I., Tamura, H., Kawachi, F., Umedzu, S. and Kinoshita, J.: Fuzzy Decision Trees by a Fuzzy ID3 Algorithm and Its Application to Diagnosis Systems, *Proceedings of the Third IEEE International Conference on Fuzzy Systems*, pp.2113-2118, 1994.
 - [15] Blake, C. L. and Merz, C. J.: UCI Repository of Machine Learning Databases [http://www.ics.uci.edu/~mllearn/MLRepository.html], University of California, Irvine, Department of Information and Computer Science, 1998.

(2005年12月18日 受付)

(2007年4月22日 採録)

[問い合わせ先]

〒599-8531 堺市中区学園町1-1

大阪府立大学 大学院理学系研究科 情報数理学専攻

松本 裕二

TEL: 072-254-9675

FAX: 072-254-9930

E-mail: matumoto@marron.cias.osakafu-u.ac.jp

著者紹介

まつもと ゆうじ
松本 裕二 [学生会員]

2001年大阪府立大学大学院総合科学研究科修士課程修了。同年、富士通株式会社入社。2002年富士通株式会社退社。同年、大阪府立大学大学院理学系研究科博士後期課程入学、現在、在学中。

うまの もとひろ
馬野 元秀 [正会員]

1974年大阪大学基礎工学部情報工学科卒業。1979年同大学院後期課程修了。工学博士。同年岡山理科大学理学部応用数学科講師。1985年大阪大学大型計算機センター助手。その後、講師、助教授を経て、1991年1月同大学工学部精密工学教室助教授、1993年10月同大学基礎工学部システム工学科助教授。1996年大阪府立大学総合科学部数理・情報科学科教授。2005年公立大学法人化により現在の所属になる。ファジィ集合論の応用、特に、データベース、知識情報処理への応用に関する研究に従事。情報処理学会、電子情報通信学会、日本ソフトウェア科学会、人工知能学会、システム制御情報学会、計測自動制御学会、ACM、IEEE各会員。

とうまる まさひろ
藤丸 雅弘 [学生会員]

2004年大阪府立大学大学院理学系研究科博士前期課程修了。同博士課程在学中。

せ た かずひさ
瀬田 和久 [非会員]

1998年大阪大学大学院工学研究科博士後期課程修了。博士(工学)。同年、大阪大学産業科学研究所助手。現在、大阪府立大学大学院理学系研究科准教授。オントロジー工学、知的学習支援システムに関する研究等に從事。人工知能学会、情報処理学会、電子情報通信学会、教育システム情報学会各会員。

Learning by Switching Knowledge Representations

by

Yuji MATSUMOTO, Motohide UMANO, Masahiro TOMARU and Kazuhisa SETA

Abstract :

When we solve a problem, we initially have no knowledge and we memorize the raw data. Finally we have general knowledge for solving the problem. To simulate this learning process, we propose a learning method by switching different levels of knowledge representations, each has a knowledge-set, a knowledge generation method and several reasoning methods. The system reasons the class of given data and the successive incorrect classifications trigger to learn with switching the reasoning method to another, reconstructing the knowledge-set, or switching the knowledge representation to the suitable one for the given data. We implement a prototype of the system and apply it to Pen-Based Recognition of Handwritten Digits in UCI Machine Learning Repository. In the simulation, the system can switch suitable knowledge representations from the raw data to general knowledge such as fuzzy rules with a good rate correctly classified and a small amount of knowledge.

Keywords : switching of knowledge representations, incremental learning, knowledge acquisition, reasoning, learning plateau

Contact Address : Yuji MATSUMOTO

*Department of Mathematics and Information Sciences, Graduate School of Science,
Osaka Prefecture University*

1-1, Gakuen-cho, Naka-ku, Sakai, Osaka 599-8531, JAPAN

TEL : 072-254-9675

FAX : 072-254-9930

E-mail : matumoto@marron.cias.osakafu-u.ac.jp