

○本堂直浩（北大工） 成瀬継太郎（北大工） 鈴木恵二（北大工） 嘉数侑昇（北大工）

要旨

与えられたデータに潜在する特徴、傾向をアルゴリズムによって抽出しようとする試みであるクラシファイヤースystemは遺伝的アルゴリズムを使用した適応的学習としてよく知られるものである。本論ではその改良型クラシファイヤースystemとして、新たに決定木の構造的特徴を取り入れた探索オペレーター、DTOを提案し、テストデータとしてMONKSを扱い計算機実験により有効性を示す。

1. はじめに

与えられた問題領域の特徴をアルゴリズムによって抽出しようとする試みである概念学習は、様々なアプローチによって行われている[1]。強力な探索能力を持つ遺伝的アルゴリズムを用いて学習を行うクラシファイヤースystem(以下CS)は、概念学習の一つのアプローチとして良く知られている[2]。CSは、学習対象の例を用いて学習する教師あり学習の中に含まれ、その与えられた例に適合するクラシファイヤーのみがビディングプロセス、遺伝的オペレーターの組み合わせによって生成するシステムである。一方、概念学習の別のアプローチとして同じ教師あり学習である決定木によるものがある。決定木によって表現される概念は、ツリー構造を通して解の構造を十分に表している。そこで本論では、CSの高い学習能力に決定木の構造的特徴を取り入れるクラシファイヤーオペレーター、*DTO*(Decision Tree Operator)を提案し、計算機実験で有効性を確認する。

2. Classifier System

CSは、クラシファイヤーと呼ばれる固定長のif-thenルールの集団を用いて行う適応的学習システムである。クラシファイヤーは次のように表現される。

$$A_1, A_2, \dots, A_n / c / s.$$

A_n は条件部であり、概念学習の場合は問題領域のそれぞれ属性が割り当てられる。 c は条件部が属すクラスを表す。また、 s は強度と呼ばれ、そのクラシファイヤーの学習対象に対する適応度を表している。以下に、CSのアルゴリズムを示す[2]。

Table 1 CS algorithm.

1. 初期集団をランダムに生成する。
2. 与えられた例に条件部が一致するクラシファイヤーを強度に応じて選択する。
3. 選択されたクラシファイヤーのクラス部を例に照会する。
4. クラス部の評価を行い、そのクラシファイヤーの強度の更新を行う。(ビディングプロセス)
5. 2~4のステップを1世代とし、確率的にある世代毎に遺伝的操作、交叉・突然変異を行う。

3. 決定木

決定木を用いる学習は、記述言語としてツリー構造を用い、情報理論に基づいて決定木を生成し概念獲得を行うものであり、QuinlanのID3[3]が良く知られている。与えられた例のルールの適合のみを議論するCSと異なり、ツリー構造を通じて表現される学習対象の概念は構造的に表現されることが決定木による学習の最大の特徴である。しかしながら決定木による学習の効率性は問題に大きく依存し、問題の構造を完全に表現することが困難である問題が存在する[4]。そのため、問題の特質によって様々な拡張が行われている(例えば、[5])。

このような性能の問題依存を抱えながらも決定木のアドバンテージがなおも存在する。まず、学習時間がCS等に比べて極端に少ないこと。また、先に述べた特徴である構造的な解表現が可能であることである。たとえ、困難な問題であっても決定木のサブツリーには解の一部が含まれていることが多く、決定木をルールの集合として考えた場合、そのような木は不完全ながらも個々のルールとしてはある程度意味を持つ。本論ではこのような決定木のアドバンテージをCSの探索オペレーターとして利用することを考える。

4. DTO: CSにおける構造的オペレーター

一般にCSにおけるオペレーターは、交叉と突然変異が使用される。交叉は、適応度(強度)の高いクラシファイヤー同士を交叉させることにより、より問題に適合するクラシファイヤーを生成しようとするものである。また突然変異は、明らかに適応度の低いクラシファイヤーの代わりにランダムに生成したクラシファイヤーと入れ替えるオペレーターである。共に、初期集団には存在しないクラシファイヤーを得ようとするものである。

本論で提案する新しいオペレーター、DTOは前出の決定木の特徴をCSに取り入れるものである。つまり、クラシファイヤーのオペレーターとして、クラシファイヤー集団から決定木を生成し、集団の傾向を構造的に表現した決定木から新しいクラシファイヤーを生成するものである。以下に、DTOのアルゴリズムを示す。

Table2 Operator DTO algorithm.

1. 集団中の一定強度以上のクラシファイヤーからID3を用いて決定木を生成する.
2. 生成された決定木からランダムにN本ルールを抽出する.
3. 得られたルールをクラシファイヤーに変換する. 初期強度は, 生成の際に使用したクラシファイヤーの強度の平均値とする.
4. 集団の中から強度の低い順にN本取りだし, 3で得られたクラシファイヤーと交換する.

1. 一定以上の強度のクラシファイヤーとは, 少なくとも初期強度以上の強度を持つものを指す. ビッティング操作において適合しないクラシファイヤーは強度が低下するため, 決定木を生成するためのクラシファイヤーとして少なくとも初期強度より小さいものは不適切であるとおわれる.

2. 決定木はルートとなる属性から, クラスを示すリーフまでのパスを一本のルールとみなすことができ, すなわち決定木はルールの集合であるといえる. ルールの抽出とは, その決定木中のパスをランダムに選択することである.

3. その取り出したルールは, 必ずしもすべての属性を含むわけではないため, 含まれない属性はワイルドカード(don't care)としてクラシファイヤーに変換する.

このようにして決定木を用いて新しいクラシファイヤーが生成される. 1の条件で生成された決定木は完全に問題の構造を表していることは稀である. しかし, 決定木のサブツリーには部分的に構造を表していることが多く, また, 決定木の特徴からデフォルトハイアラーキーが表現しやすいため, CSでは学習が困難なルールも決定木を通じて獲得が可能である.

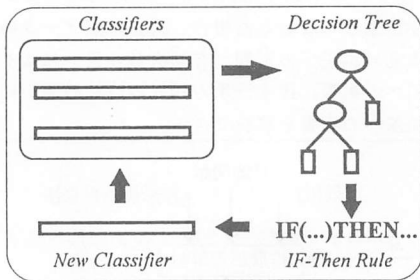


Fig.1 Operator DTO.

5. 計算機実験

本論で扱う問題は, MONKS-1[1]である. MONKS-1は, Thrunによるテストデータ群であり, 概念学習アルゴリズムの性能評価データとして良く知られている. MONKS-1は6個の属性からなり, 獲得概念は(属性1=属性2)または(属性5=0)であるときクラスがポジティブであるものである. CSが学習する際にはそれぞれの属性はクラシファイヤーの遺伝子座に, クラスは行動部に割り当てられる.

提案手法であるDTOと他のオペレーターの比較として, CSにおける学習過程に両者を同様の確率でそれぞれを独立に, 更に複合的に行う場合について実験を行う. CSの詳細なセッティングはWilsonのCSに準ずる[2].

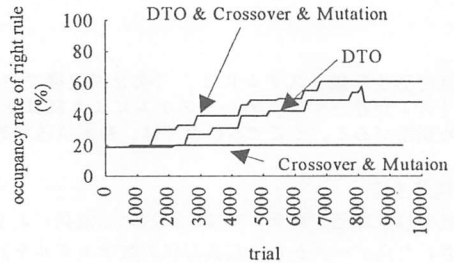


Fig.2 Result.

結果は, Fig. 2に示される. y軸はクラシファイヤー集団内における学習対象の概念を含むクラシファイヤーの占有率である. 学習過程において交叉・突然変異を行った場合, ほぼ初期集団に含まれていたクラシファイヤーしか獲得しなかった. DTOを行った場合, 有効なクラシファイヤーがより早い段階で得られている. 最も良い結果を得たのは, 両者を同時に使用した場合である. 集団内の傾向をID3によって決定木を生成し, そこから新しいクラシファイヤーを得ることにより探索速度が早まることが確認できた.

6. おわりに

決定木を利用してクラシファイヤーの集団の中から新しいクラシファイヤーを生成するオペレーターを提案し, 計算機実験で有効性を確認した.

・参考文献

- [1] Sebastian Thrun. (1992). The MONKS's problems - A Performance Comparison of Different Learning algorithm. *Technical Report CS-CMU-91-197, Carnegie Mellon University in Dec.*
- [2] Wilson, S. W.(1987). Classifier systems and the animat problem. *Machine Learning 2, 3, 199-228.*
- [3] Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning 1, 1, 81-106.*
- [4] Quinlan, J. R. (1993). C4.5 Programs for Machine Learning. *Morgan Kaufmann Publishers 2929 Campus Drive, Suite 260 San Mateo, CA94403.*
- [5] R. López de Mántaras. (1991). A Distance - Based Attribute Selection Measure Decision Tree Induction. *Machine Learning, 6, 81-92.*