유전자 알고리즘 기반의 불완전 데이터 학습을 위한 속성값계층구조의 생성

주 진 우[†] · 양 지 훈[††]

요 약

부부분불완전 데이터(Partially Missing Data) 또는 데이터의 속성 값이 표현되는 정도의 깊이가 서로 다른 데이터를 학습하는데 있어서 속성값계층구조(Attribute Value Taxonomy, AVT)를 기반으로 학습하면 기존의 학습 알고리즘을 통해 얻은 결과보다 정확하고 간결한 분류기를 얻을 수 있다는 사실이 밝혀졌다. 하지만 이러한 속성값계층구조는 처음부터 전문가 또는 데이터 도메인에 대한 지식을 가지고 있는 사람에 의해 만들어져 제공되어야 한다. 이러한 수작업을 통한 속성값계층구조를 생성하기 위해서는 많은 시간이 걸리며 생성과정에서 오류가 발생할 수 있다. 또한 데이터 도메인에 따라서 속성값계층구조를 제공할 전문가가 부재한 경우가 있다. 이러한 배경 아래 본 논문은 유전자 알고리즘을 통해 자동으로 근 최적의 속성값계층구조를 생성하는 알고리즘(GA-AVT-Learner)을 제안한다. 본 논문의 실험은 다양한 실제 데이터를 가지고 GA-AVT-Learner로 생성한 속성값계층구조를 다른 속성값계층구조와 비교하였다. 따라서 GA-AVT-Learner에 의해 생성된 속성값계층구조가 정확하고 간결한 분류기를 제공함을 보이고, 불완전데이터 처리에 있어서도 높은 효율을 보임을 실험적으로 증명하였다.

키워드 : 속성값계층구조, 의사결정분류기, 유전자 알고리즘

Genetic Algorithm Based Attribute Value Taxonomy Generation for Learning Classifiers with Missing Data

Jinu Joo[†] · Jihoon Yang[††]

Abstract

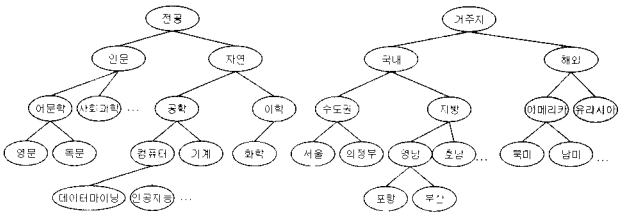
Learning with *Attribute Value Taxonomies (AVT)* has shown that it is possible to construct accurate, compact and robust classifiers from a partially missing dataset (dataset that contains attribute values specified with different level of precision). Yet, in many cases AVTs are generated from experts or people with specialized knowledge in their domain. Unfortunately these user-provided AVTs can be time-consuming to construct and misguided during the AVT building process. Moreover experts are occasionally unavailable to provide an AVT for a particular domain. Against these backgrounds, this paper introduces an AVT generating method called GA-AVT-Learner, which finds a near optimal AVT with a given training dataset using a genetic algorithm. This paper conducted experiments generating AVTs through GA-AVT-Learner with a variety of real world datasets. We compared these AVTs with other types of AVTs such as HAC-AVTs and user-provided AVTs. Through the experiments we have proved that GA-AVT-Learner provides AVTs that yield more accurate and compact classifiers and improve performance in learning missing data.

Key Words : Attribute Value Taxonomy, Decision Tree Classifier, Genetic Algorithm

1. 서 론

불완전데이터를 학습하는 방법은 어려운 문제로 여겨왔고 많은 방법이 제시되었다[1, 2, 11, 12]. 그 중에서도 부분 불완전데이터(Partially Missing Data)를 처리하는 방법인 AVT-DTL 알고리즘이 최근 제시되었다[3]. 부분불완전데이터란 데이터의 값 일부가 추상적인 값으로 매겨진 데이터를 통칭한다. 즉, 데이터의 값들의 정밀도 단계가 서로 다른 데이터를 부분불완전 데이터라고 말하며 실제로 데이터를 수집하는 과정에서 빈번히 발생한다. AVT-DTL이 불완전데이터를 처리하기 위해서는 사용자 또는 전문가에 의해 제공된 속성값계층구조(Attribute Value Taxonomy)를 요구한다. 속성값계층구조는 부분불완전데이터를 통하여 의사결정분류기(Decision Tree Classifier)를 학습하는데 있어서 지침 역할을 한다. 속성값계층구조란 속성값의 정밀도에 따라 속성값을 계층적인 연관성을 지닌 트리의 형태로 재구성한 일종의 온톨로지(Ontology)이다. (그림 1)은 전공, 거주지에 대한 속성값계층구조를 나타낸 것이다. 데이터마이닝과 인공지능은 컴퓨터의

※ 본 연구는 한국과학재단 특정기초연구 (R01-2004-000-10689-0) 지원으로 수행되었음
† 준 회 원 : LG전자 MC사업본부 단말연구소 연구원
†† 정 회 원 : 서강대학교 컴퓨터학과 부교수(교신저자)
논문접수 : 2006년 1월 13일, 심사완료 : 2006년 3월 22일



(그림 1) 전공 및 거주지 속성에 대한 속성값계층구조

한 분야이고 컴퓨터와 기계는 공학 분야에 속하기 때문에 아래와 같은 모양의 속성값계층구조를 구성할 수 있다.

이러한 속성값계층구조는 사용자나 전문가에 의해 직접 작성해야하므로 시간과 노력이 많이 들며 정확도면에서 오류가 발생할 수 있다. 본 논문은 속성값계층구조를 자동으로 생성하는 GA-AVT-Learner를 제안하여 이러한 시간과 노력을 줄이고 보다 정확한 속성값계층구조를 생성하는데 목표를 두었다. GA-AVT-Learner는 유전자 알고리즘[4]을 기반으로 여러 세대에 걸쳐 해를 진화 시키는 탐색 과정을 통해 근 최적의 속성값계층구조를 찾는 알고리즘이다.

본 논문의 구성은 다음과 같다. 2장에서는 AVT-DTL, 유전자 알고리즘, 그리고 각종 관련 연구에 대해 살펴본다. 3장에서는 본 논문이 제안한 GA-AVT-Learner에 대해 소개한다. 4장에서는 다양한 실제 데이터를 바탕으로 GA-AVT-Learner로 학습한 속성값계층구조를 통해 생성되는 분류기의 정확도와 간결성을 통해 속성값계층구조의 성능을 비교, 분석하였다. 끝으로 5장에서는 본 연구의 요약과 앞으로의 개선방안 등에 대해 논하였다.

2. 관련연구

속성값계층구조(Attribute Value Taxonomy, AVT)[3]란 하나의 속성(Attribute)이 가질 수 있는 속성 값(Attribute Value)들의 계층적인 관련성에 따라 트리의 형태로 재구성한 자료구조이다. 속성값계층구조 τ_i는 내부의 연결된 두 노드는 ISA 릴레이션을 가지면서, τ_i의 속성값들의 추상적 표현의 정밀도에 대한 계층적 관계를 나타낸 자료구조이다.

속성값계층구조 기반 의사결정분류기 학습기(Attribute Value Taxonomy guided Decision Tree Learner, 이하 AVT-DTL)는 불완전 데이터 학습에 있어서 우수한 성능을 발휘하는 학습 알고리즘이다. AVT-DTL 알고리즘은 속성값계층구조를 통해 불완전데이터를 처리하여 정확하고 간결한 의사결정 분류기를 탐색하는 학습알고리즘이다[3]. 의사결정분류기를 생성하는 과정은 일반적인 의사결정분류기 학습알고리즘과 유사하지만 데이터를 분류하는 과정에서 기준으로 삼는 속성값이 속성값계층구조에 의해 결정된다. 즉, 데이터를 분할 할 때 기준이 되는 속성값들은 속성값계층구조의 내부노드들을 기준으로 삼는다. 따라서 AVT-DTL은 데이터 분할 과정에서 인포메이션 게인(Information Gain)을 가장 높이는 속성 선택 뿐만 아니라 그 속성이 가지고 있는 속성값계층구조의 레벨도 함께 고려해야한다.

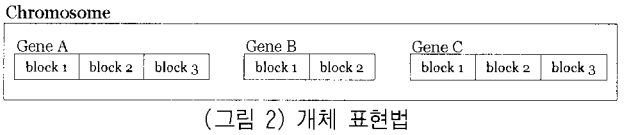
계층적 응집 클러스터링 기반 속성값계층구조 학습기(Hierarchical Agglomerative Clustering Attribute Value Taxonomy Learner, 이하 HAC-AVT-Learner)[5]는 각 속성값에 대한 클래스의 분포를 기준으로 계층적 응집 클러스터링(Hierarchical Agglomerative Clustering)을 사용하여 유사한 속성값을 클러스터링 함으로써 속성값계층구조를 생성하는 알고리즘이다. 속성값간의 유사도는 옌센-샤논 다이버전스[6]를 이용한다.

유전자 알고리즘(Genetic Algorithm, GA)은 발견적 계산 모델로 탐색공간에서 최적 해를 찾는데 문제에 이용된다. 효율적으로 해를 찾기 위해서 해를 유전자 연산에 적합한 형태의 자료구조로 부호화하고 초기화된 임의의 후보 해 집단에서 출발하여 적합도(Fitness) 값에 따라 유전자 연산자(Genetic Operator)에 의해 확률적으로 개체의 일부 혹은 전체가 다음 세대로 유전되면서 점점 해 집단이 최적의 해로 수렴해간다[4, 7, 8].

3. 유전자 알고리즘 기반 속성값계층구조 생성기

3.1 개체 표현법

유전자 알고리즘을 이용한 탐색을 위해서는 속성값계층구조의 적절한 개체 표현법(Representation)이 필요하다. 일반적인 트리 자료구조로 개체를 표현할 경우 다음과 같은 두 가지 문제를 지니고 있다. 첫째, 교배 연산 수행 후 중복되는 노드가 발생한다. 둘째, 진화 연산 후 말단노드가 변한다. 이러한 문제를 보완하는 정수를 이용한 이진 트리 구조의 개체 표현법을 (그림 2)과 같이 제안한다[9].



(그림 2) 개체 표현법

하나의 개체는 데이터의 모든 속성에 대한 속성값계층구조를 함축하고 있다. 이러한 개체를 염색체(Chromosome)라고 지칭하며 염색체는 하나의 속성에 대한 속성값계층구조를 나타내는 유전자(Gene)를 여러 개 가지고 있다. i번째 속성의 초기 속성값들의 집합을 V_i라고 할 때 하나의 유전자가 가지는 블록(Block)의 개수는 $|V_i|-2$이다. 블록은 속성값계층구조가 트리의 각 단계에서 서브트리를 형성하는 두 노드의 인덱스로 나타낸다. 개체 표현법의 디코딩에 앞서 다음과 같이 컷을 정의한다[13].

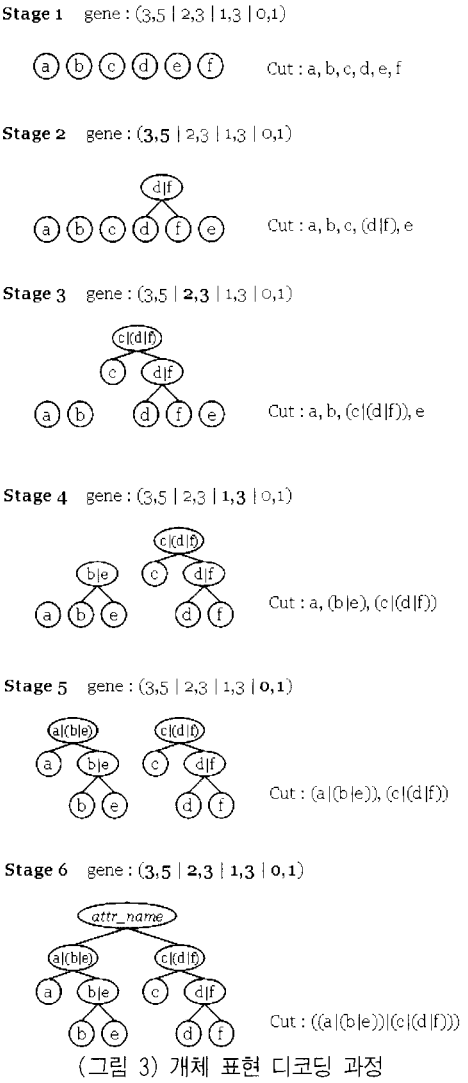
[정의 1] $\Theta(i)$를 i번째 속성값계층구조의 말단노드에 해당하는 속성값의 집합, $\Lambda(v)$를 노드 v의 조상노드, $\Psi(v)$를 노드 v의 자손 노드라고 할 때 ,i번째 속성값계층구조에 대한 컷(Cut) Γ_i는 i번째 속성값계층구조 노드들의 전체집합의 부분집합으로 다음과 같은 두 속성을 만족한다.

(1) $\forall l \in \Theta(i),\ l \in \Gamma_i$ or $(l \in \Psi(v) \wedge v \in \Gamma_i)$

(2) $\forall v,u \in \Gamma_i\ (v \neq u),\ v \notin \Psi(u)\ \wedge\ u \notin \Lambda(v)$

이러한 표현법을 통해 속성값계층구조를 디코딩 하는 과정은 다음과 같다. 첫째, 속성 i에 대한 초기 속성값으로 Γ_i를 초기화 한다. 둘째, 첫 번째 블록부터 시작하여 블록안의 두 정수를 인덱스로 하는 Γ_i의 두 노드를 묶는 서브트리를 생성한다. 셋째, 서브트리의 부모노드를 Γ_i에 추가하고 서브트리의 두 자식노드는 Γ_i에서 제거하여 새로운 컷을 얻는다. 위 과정을 $|\Gamma_i|=1$이 될 때까지 각 블록에 대해 반복 실행한다.

예를 들어 초기 속성값 a,b,c,d,e,f 을 가지는 속성 i에 대한 개체 표현이 '3, 5 | 2, 3 | 1, 3 | 0, 1' 일 때 (그림 3)과 같이 속성값계층구조를 유도할 수 있다. 우선 초기 속성값으로 컷을 초기화한다. 첫 번째 블록 (3, 5)에 의해 컷의 세 번째 와 다섯 번째 속성값 d,f를 자식 노드로 하는 서브트리를 만든다. 서브트리의 부모노드 $(d|f)$를 컷에 추가하고, 서브트리를 구성한 두 원소 d,f를 컷에서 제거한다. 이러한 과정을 컷의 원소가 하나 남을 때 까지 모든 블록에 대해서 반복 수행한다.



(그림 3) 개체 표현 디코딩 과정

3.2 유전 연산자

GA-AVT-Learner 내부의 유전자 연산자는 다음과 같이 정의한다. 우선 다음 세대에 직접전이 또는 교배를 위해 개체를 선택하는 선택 연산자로는 룰렛휠 선택 연산자(Roulette Wheel Selection Operation)를 이용하였다. 룰렛휠 선택연산자는 개체의 평가값에 비례하는 확률로 임의의 개체를 샘플링하는 방법이다.

둘째, 재조합 연산자는 기본적으로 한점교배 연산자(One-Point Crossover)를 이용하였으나 한 개체는 여러 속성값계층구조를 표현하고 있다. 따라서 각 속성값계층구조를 나타내는 유전자에 대해 각각 한점교배 연산을 수행하여 결국 염색체 전체에 대해서는 n점 교배 연산을 수행하였다.

셋째, 돌연변이 연산자는 일정한 돌연변이 확률 p_m으로 개체의 일부분을 랜덤하게 선택한 정수 값으로 치환한다. 이때, 치환되는 새 정수는 컷의 정수 인덱스 범위 내에서 이루어져야한다.

재조합 연산과 돌연변이 연산 과정 후 생성된 새로운 개체는 반드시 개체 부호화 적합성을 검사한 후 다음 세대로 보내어진다. 개체 부호화 적합성 검사란 새로 생성된 개체가 속성값계층구조로 디코딩 될 수 있는 개체 표현인지를 검사하는 것으로써 한 개체는 다음과 같은 두 가지 특성을 만족해야한다. 첫째, 각 블록안의 정수는 그 단계 컷의 최대 인덱스 값을 넘지 않는다. 둘째, 한 블록안의 두 정수는 같지 않다.

마지막으로 매 세대마다 생성되는 개체군을 평가하여 평가값을 매기는 평가함수(Fitness Function)가 필요하다. 속성값계층구조의 일차적인 목표는 정확하고 간결한 의사결정분류기를 생성하는 것이다. 따라서 평가함수도 이러한 관점에 맞추어 각 속성값계층구조를 통해 AVT-DTL을 수행하여 얻어진 의사결정분류기의 정확도와 크기에 각각 일정한 가중치를 둔 식 (1)을 사용하였다.

$$Fitness(x_i) = w_{ac}\,Acc(x_i) + w_{ts}\,Ts(x_i) \qquad (1)$$

이 식은 개체 x_i에 대한 평가함수로 $Acc(x_i)$와 $Ts(x_i)$는 의사결정분류기의 정확도와 크기를 각각 나타낸 함수이며, w_{ac}와 $w_{ts}=1-w_{ac}$는 이에 대한 가중치이다.

3.3 알고리즘

유전자 알고리즘 기반 속성값계층구조 학습기(Genetic Algorithm Attribute Value Taxonomy Learner, GA-AVT-Learner)는 유전자 알고리즘에 기초하여 속성값계층구조 후보군을 진화 시키는 방법이다. 진화 과정에서 매 세대마다 후보군의 각 개체를 평가하는 수단은 속성값계층구조 기반 의사결정트리 학습기(AVT-DTL)를 이용하며, 이를 통해 생성된 의사결정분류기의 정확도와 분류기의 크기에 가중치를 주어 생성한 평가 값을 통해 속성값계층구조 후보들을 평가한다. 이러한 평가방법은 학습알고리즘에 의존적으로 속성값계층구조를 생성하는 방법으로서 래퍼접근법(Wrapper Approach)이라 한다[10]. 래퍼접근법은 학습알고리즘 자체에 의존하기 때문에 계산 시간은 길지만 학습알고리즘에 더욱 적합한 결

과를 도출해낼 수 있다는 장점이 있다. 이와 반대의 방법이 필터접근법(Filter Approach)으로 일반적으로 계산시간은 래퍼접근법보다 빠르지만 래퍼접근법에 비해 학습알고리즘에 적합하지 않은 결과를 도출할 가능성이 있다. HAC-AVT-Learner는 필터접근법을 따른다.

[알고리즘 1] GA-AVT-Learner

Input : Dataset $S \subseteq V_1 \times V_2 \times ... \times C$,
Fitness Function $f_{AVT-DTL}()$
Genotype of AVTs $X = \{X_1, X_2, ..., X_n\}$

Output : Best AVT $T^* = \{T_1^*, T_2^*, ..., T_n^*\}$
$T^* = argmax_{T \subset AVT}(f_{AVT-DTL}(T))$

Step 1. Initialize $X = \{X_1, X_2, ..., X_n\}$

Step 2. Do Until termination condition is satisfied :
(1) election operation :
$X_S = selection(X)$
(where $selection()$ is selection process from parent population)
(2) crossover operation :
$X_C = crossover(X_S)$
(where $crossover()$ is the crossover operation with a crossover probability)
(3) mutation operation :
$X_M = mutation(X_C)$
(where $mutation()$ is the mutation operation)
(4) reserve elite chromosome :
$X_N = X_M \cup X_E$
(where X_E is the elite chromosomes)
(5) evaluate fitness value :
with fitness function $f_{AVT-DTL}(X_N)$

따라서 우리는 GA-AVT-Learner를 통해 HAC-AVT-Learner의 한계를 극복하고 근 최적의 속성값계층구조를 발견하여 보다 정확하고 간결한 의사결정분류기를 생성하고자 한다. GA-AVT-Learner의 수행과정은 [알고리즘 1]과 같다.

4. 실험 및 결과

4.1 실험 데이터

본 실험은 총 여덟 개의 실제 데이터를 통해 실험을 수행하였다. 모든 데이터의 속성 값들은 노미널(Nominal) 값을 가지며, 사용자 제공 속성값계층구조가 제공된 데이터를 선택하였다. 모든 실험데이터는 UCI Machine Learning Repository[1] 의 컴퓨터학습 공개 도메인에서 가져온 실제 데이터이다. 두 번째 실험에서 사용된 0% ~ 50%의 전불완전데이터는 완전 데이터에서 해당하는 퍼센트 비율만큼의 속성 값들을 "?"로 변환하여 생성한 데이터이다.

1) http://www.ics.uci.edu/~mlearn/MLRepository.html

4.2 실험 방법

본 논문의 실험은 크게 두 가지로 이루어진다. 첫 째, 여러 실제 데이터로부터 GA-AVT-Learner를 이용하여 속성값계층구조를 생성하는 실험을 통해 GA-AVT-Learner의 일반적인 성능을 측정하고자 하였다. 둘 째, 0% ~ 50%의 비율로 데이터의 값이 부재한 전불완전 데이터를 통해 각 속성값계층구조의 성능을 평가하였다.

GA-AVT-Learner를 평가하는 비교대상 알고리즘은 다음 세 가지로 선정하였다. 첫 번째는 속성값계층구조를 사용하지 않은 경우로 C4.5 알고리즘을 통해 불완전 데이터를 학습하였다. 두 번째는 사용자제공 속성값계층구조를 이용한 AVT-DTL 알고리즘과 비교하였다. 마지막으로 HAC-AVT-Learner로 학습한 속성값계층구조를 이용한 AVT-DTL과 비교하였다. 첫 번째 알고리즘인 C4.5를 제외한 나머지 세 개는 AVT-DTL을 분류기 학습알고리즘으로 사용하였다.

두 실험 모두 전체 데이터의 1/3을 속성값계층구조를 학습하는 트레이닝 데이터로 사용하였고, 나머지 2/3의 데이터를 의사결정분류기를 학습하는데 사용함으로써 속성값계층구조를 생성하는 데이터와 평가하는 데이터 간에 정보의 유출이 없도록 하였다. 결국 하나의 데이터에 대한 최종 평가값은 세 개의 트레이닝 데이터를 통해 얻은 평가 값의 평균이다.

각 알고리즘의 성능 평가 기준은 속성값계층구조로 학습한 의사결정분류기의 정확도(Accuracy)와 의사결정분류기의 크기(Tree Size), 그리고 의사결정분류기의 말단노드의 크기(즉, 전체 룰의 크기, Node Size)이다. 의사결정분류기의 정확도는 테스트 데이터에 대해 10-fold cross validation을 수행한 정확도이다. GA-AVT-Learner의 파라미터는 다음과 같이 설정하였다 :

- 교배확률(Crossover Probability) : 선택된 두 개체가 재조합 연산을 수행할 확률이다. 이 실험에서는 50%로 설정하였다.
- 돌연변이확률(Mutation Probability) : 개체에 돌연변이가 생성될 확률로써 2% 로 설정하였다.
- 개체군의 수(Population) : 한 세대에 존재하는 후보 속성값계층구조의 수로 50개로 한정하였다.
- 진화세대 수(Generation) : 진화세대의 총수로 본 실험에서는 100 세대로 설정하였다.
- 정지정책(Stoping Criteria) : 정렬된 개체군의 평가치가 높은 개체부터 90%가 하나의 해로 수렴하면 GA-AVT-Learner를 종료한다.

4.3 실험 결과

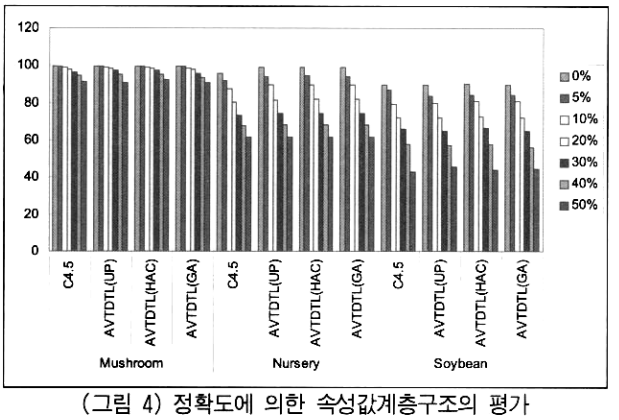
첫 번째 실험 결과를 보여주는 <표 1>을 통해 볼 때 각 알고리즘의 정확도에서는 크게 차이를 보이지 않았다. 반면 의사결정분류기의 크기와 말단노드의 수에서는 눈에 띄는 차이를 관찰할 수 있었다. 각 데이터를 살펴보면 Audiology, Car, Dermatology, Mushroom, Nursery 데이터에서 GA-AVT-Learner를 통해 생성한 속성값계층구조를 AVT-DTL에 적

용시켰을 때 가장 간결한 분류규칙을 유도하는 것을 관찰하였다. 반면, Breast, Zoo, Soybean 데이터에서는 차이를 보이지 않거나 오히려 좋지 않은 성능을 보여주었다. Zoo 데이터의 경우 C4.5의 결과를 제외하고는 세 가지 경우 모두 같은 값을 가지고 있다. Breast 데이터에서는 사용자 제공 속성값계층구조의 결과가 가장 좋게 나타났다. 이러한 결과가 나오는 이유는 속성값계층구조를 학습하기 위한 데이터의 크기가 충분하지 않아 최적의 속성값계층구조를 학습하는 트레이닝 데이터가 포괄하는 정보가 제한적이었기 때문이라고 추정된다.

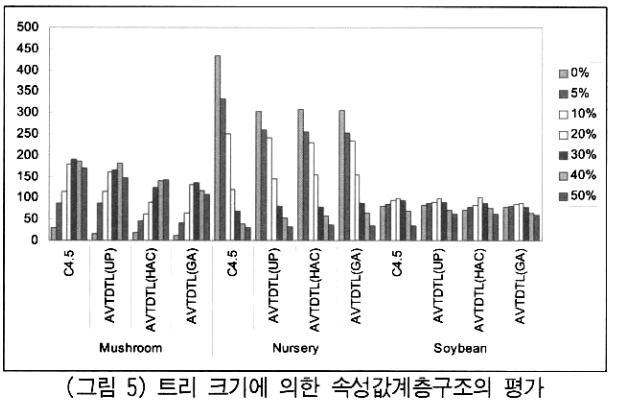
결론적으로 C4.5(즉, 속성값계층구조를 사용하지 않은)의 결과가 가장 복잡한 의사결정분류기를 만들어낸다. 반면, 속성값계층구조를 사용한 나머지 세 개의 알고리즘은 의사결정분류기의 크기를 절반정도로 줄여주는 것을 볼 수 있다. 전반적으로 사용자 제공 속성값계층구조 보다는 HAC-AVT가, HAC-AVT 보다는 GA-AVT가 보다 간결한 분류규칙을 생성하는 것을 발견하였다.

〈표 1〉 데이터별 속성값계층구조의 평가

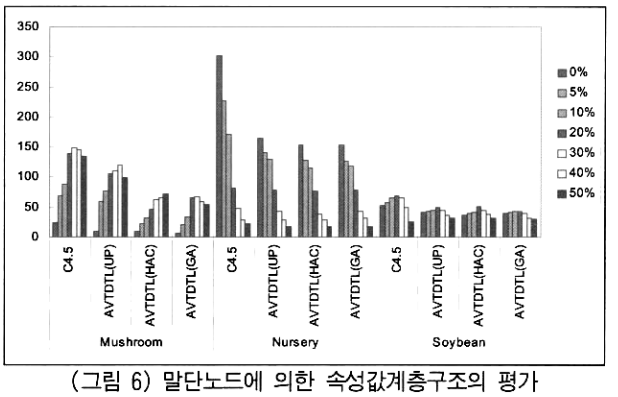
		Accuracy	Tree Size	Num. of Leaves
Audiology	C4.5	**75.7**	38.3	24.0
	AVTDTL(UP)	**75.7**	38.3	19.7
	AVTDTL(HAC)	**75.7**	37.7	19.3
	AVTDTL(GA)	72.9	**37.0**	**19.0**
Breast	C4.5	72.9	11.7	8.7
	AVTDTL(UP)	72.2	**10.3**	**5.7**
	AVTDTL(HAC)	**73.9**	20.3	10.7
	AVTDTL(GA)	69.8	14.3	7.7
Car	C4.5	89.8	118.3	85.7
	AVTDTL(UP)	**95.8**	101.7	51.3
	AVTDTL(HAC)	95.7	89.7	45.3
	AVTDTL(GA)	95.5	**74.3**	**37.7**
Dermatology	C4.5	92.1	25.0	21.0
	AVTDTL(UP)	89.4	25.0	14.7
	AVTDTL(HAC)	**94.4**	25.0	10.7
	AVTDTL(GA)	94.3	**19.0**	**9.7**
Mushroom	C4.5	**99.9**	29.0	24.0
	AVTDTL(UP)	**99.9**	16.0	10.3
	AVTDTL(HAC)	**99.9**	17.7	9.3
	AVTDTL(GA)	**99.9**	**12.3**	**6.7**
Nursery	C4.5	95.8	434.0	302.0
	AVTDTL(UP)	98.9	302.0	164.3
	AVTDTL(HAC)	**99.0**	307.0	154.0
	AVTDTL(GA)	98.9	**305.0**	**153.0**
Soybean	C4.5	89.5	79.7	52.3
	AVTDTL(UP)	89.7	81.7	41.3
	AVTDTL(HAC)	**90.1**	**71.7**	**36.3**
	AVTDTL(GA)	89.7	77.7	39.3
Zoo	C4.5	92.1	15.7	9.0
	AVTDTL(UP)	**93.1**	**13.0**	**7.0**
	AVTDTL(HAC)	**93.1**	**13.0**	**7.0**
	AVTDTL(GA)	**93.1**	**13.0**	**7.0**



(그림 4) 정확도에 의한 속성값계층구조의 평가



(그림 5) 트리 크기에 의한 속성값계층구조의 평가



(그림 6) 말단노드에 의한 속성값계층구조의 평가

(그림 6) 말단노드에 의한 속성값계층구조의 평가

두 번째 실험에서는 0% ~ 50%의 전불완전 데이터를 통해 (그림 4), (그림 5), (그림 6)에서 보이는 결과를 도출하였다. (그림 4)를 보면 대체로 네 개의 알고리즘이 비슷한 정확도를 보이는 것을 알 수 있다. 특히, Mushroom 데이터의 0%, Nursery 데이터의 10%, 30%, 그리고 Soybean 데이터의 50% 에 대해서는 GA-AVT-Learner로 학습한 속성값계층구조가 HAC-AVT-Learner로 생성한 속성값계층구조보다 높은 정확도의 분류기를 유도하는 것을 알 수 있다. (그림 5)는 전불완전 데이터의 각 퍼센트별 트리의 크기를 기술하였다. 특히 Mushroom 데이터의 0%, 5%, 40%, 50%, Nursery 데이터의 0%, 5%, 50%, Soybean 데이터의 20%, 30%, 40%, 50%에 대해서 GA-AVT-Learner로 생성한 속성값계층구조가 HAC-AVT-Learner로 생성한 속성값계층구조 보다 좋은 성

능을 보이고 있다. 끝으로 (그림 6)을 통해, 생성되는 룰의 개수에서도 GA-AVT-Learner로 생성한 속성값계층구조가 HAC-AVT-Learner로 생성한 속성값계층구조에 비해 현저히 적거나 비슷한 수치를 보임을 알 수 있다. 따라서 전불완전데이터의 학습에 있어서도 GA-AVT-Learner로 학습한 속성값계층구조가 정확하고 간결한 의사결정분류기를 유도하는 것을 알 수 있다.

5. 결론 및 향후과제

본 논문은 유전자 알고리즘을 통하여, 불완전데이터를 효율적으로 학습하는 AVT-DTL 알고리즘에 이용되는 속성값계층구조를 자동으로 생성하는 방법을 제안하였다. 본 논문에서 제안한 GA-AVT-Learner를 통해 생성된 속성값계층구조는 사용자 제공 속성값계층구조와 HAC-AVT-Learner로 생성한 속성값계층구조와 함께 비교하였다. 비교 척도는 이들 속성값계층구조를 통해 학습한 의사결정분류기의 정확도와 간결성을 기준으로 비교 평가하였다. 구체적으로 크게 다음과 같은 두 가지 실험을 통해 GA-AVT-Learner의 효율성을 입증하였다.

이후 향후과제로 다음과 같은 일들을 생각 해 볼 수 있다. 첫 째, 이진트리 형태의 속성값계층구조를 보다 일반적인 트리형태로 학습할 수 있도록 한다. 둘째, 새로운 형태의 속성값계층구조 평가함수를 찾는다. 셋째, 새로운 평가함수를 제시함으로써 진화과정을 개별 속성에 대해 따로 수행하는 방법을 제시한다. 넷째, 속성값계층구조를 다른 기계학습 알고리즘(예, Naive Bayes Learner)에 대해서도 위의 실험을 수행해볼 수 있을 것이다[13]. 끝으로 보다 다양한 데이터에 대해 GA-AVT-Learner를 적용해 봄으로써 좀더 높은 신뢰도를 가지는 결과를 도출해 낼 것으로 기대된다.

참 고 문 헌

[1] Quinlan, R., C4.5: Programs for Machine Learning : Morgan Kaufmann, San Mateo, CA, pp.27-33, 1992.

[2] Quinlan, R., "Introduction of Decision Trees," In *Machine Learning,* Vol.1, No.1, pp.81-106, 1986.

[3] Zhang, J., Honavar, V., "Learning Decision Tree Classifiers from Attribute Value Taxonomies and Partially Specified Data," *Proceedings of the Twentieth International Conference on Machine Learning (ICML 2003),* pp.880-887, 2003.

[4] Mitchell, M., Introduction to Genetic Algorithms : MIT Press, Cambridge, MA, 1996.

[5] Kang, D.K., Silvescu, A., Zhang, J., and Honavar, V., "Generation of Attribute Value Taxonomies from Data for Data-Driven Construction of Accurate and Compact Classifiers," *Proceedings of the Fourth IEEE International Conference on Data Mining (ICDM 2004),* Brighton, UK, pp. 130 137, 2004.

[6] Fuglede, B., Topsoe, F., "Jensen-Shannon Divergence and Hilbert Space Embedding," *Proceedings of the International Symposium on Information Theory (ISIT 2004),* Chicago, IL USA, p.31, 2004.

[7] Goldberg, D., Genetic Algorithm in Search, Optimization, and Machine Learning : Addison-Wesley, New York, 1989.

[8] Gen, M., Cheng, R., Genetic Algorithms and Engineering Optimization : John Wiley & Sons, Inc., Chapter 3, pp.97-141, 2000.

[9] Joo, J., Zhang, J., Yang, J., and Honavar, V., "Generating AVTs Using GA for Learning Decision Tree Classifiers with Missing Data," In *Proceedings of the Seventh International Conference on Discovery Science (DS'04),* Padova, Italy, pp. 347-354, 2004.

[10] Yang, J., Honavar, V., "Feature Subset Selection Using A Genetic Algorithm," In *Feature Extraction, Construction and Selection - A Data Mining Perspective,* Motoda and Liu (ed.), Kluwer Academic Publishers, Chapter 8, pp.117-136, 1998.

[11] Mitchell, T., Machine Learning : McGraw-Hill Companies, Inc., Chapter 3, pp.52-80, 1997.

[12] Duda, R., Hart, P., and Stork, D., Pattern Classification second edition : Wiley-interscience, Inc., Chapter 8, pp.409-413, 2000.

[13] Zhang, J., Honavar, V., "AVT-NBL: An Algorithm for Learning Compact and Accurate Naive Bayes Classifiers from Attribute Value Taxonomies and Data," In *Proceedings of the Fourth IEEE International Conference on Data Mining (ICDM 2004),* Brighton, UK, pp.289-296, 2004.

주 진 우

e-mail : jujoo@lge.com
2004년 서강대학교 컴퓨터학과(학사)
2006년 서강대학교 컴퓨터학과 (공학석사)
2006년~현재 LG전자 MC사업본부 단말연구소 연구원
관심분야 : 기계학습, 데이터마이닝, 인공지능, 진화알고리즘 등

양 지 훈

e-mail : yangjh@sogang.ac.kr
1987년 서강대학교 전자계산학과(학사)
1989년 아이오와 주립대학교 대학원 컴퓨터학과(공학석사)
1999년 아이오와 주립대학교 대학원 컴퓨터학과(공학박사)
1999년~2000년 HRL Laboratories, LLC., Research Staff Member
2000년~2002년 SRA International, Inc., Professional Staff Member
2002년~현재 서강대학교 컴퓨터학과 부교수
관심분야 : 기계학습, 데이터마이닝, 인공지능, 생물정보학 등