유전 알고리즘 기반 귀납적 학습 환경을 위한 건설적 귀납법

김 영 준*

Constructive Induction for a GA-based Inductive Learning Environment

Yeongjoon Kim*

이 논문은 2006년도 상명대학교 소프트웨어 · 미디어연구소 학술연구비 지원으로 수행되었음

요 약

건설적 귀납법은 사례들이 갖고 있는 속성들에 적합한 연산자를 적용하여 이들 사례들을 좀 더 효율적으로 분류할 수 있는 새로운 속성들을 도출해 내는 기법이다. 본 논문에서는 주어진 사례의 집합으로부터 PROSPECTOR에서 사용한 규칙 형태의 분류 규칙을 습득하는 유전 알고리즘 기반 귀납적 학습 환경을 위한 건설적 귀납법을 제시한다. 속성 결합 연산자와 유도된 속성의 유용성을 평가하기 위한 방법을 중심으로 건설적 귀납법에 대해 자세히 설명하고 다양한 사례 집합을 이용하여 건설적 귀납법이 유전 알고리즘 기반 학습 환경에 미치는 영향을 평가하였다.

ABSTRACT

Constructive induction is a technique to draw useful attributes from given primitive attributes to classify given examples more efficiently. Useful attributes are obtained from given primitive attributes by applying appropriate operators to them. The paper proposes a constructive induction approach for a GA-based inductive learning environment that learns classification rules that are similar to rules used in PROSPECTOR from given examples. The paper explains our constructive induction approach in details, centering on operators to combine primitive attributes and methods to evaluate the usefulness of derived attributes, and presents the results of various experiments performed to evaluate the effect of our constructive induction approach on the GA-based learning environment.

키워드

건설적 귀납법, 유전 알고리즘, 귀납적 학습, PROSPECTOR

I. 서 론

유전 알고리즘[1]은 주어진 문제에 대하여 이진 문자를 이용하여 코딩 된 가능한 해들로 개체 집단을 생성한 후 개체 집단내의 구성원에 생물학적 진화 과정에서 볼 수 있는 유전 연산자들을 적용하여 새로운 개체 집단을 생성하는 과정을 반복하면서 주어진 문제의 최적 해를 찾는 탐색 알고리즘이다. 유전 알고리즘은 일반적인 탐색 문제 및 여러 최적화 문제 등의 해결에 널리 이용되어 왔으며 기계 학습 분야에서는 다양한 학습 시스템의 구축과 함께 생성 규칙의 습득[2], 퍼지 컨트롤러와 분류 시스템의 구현을 위한 퍼지 규칙의 습득[3][4] 등에 이용

* 상명대학교 소프트웨어학부

접수일자 : 2006. 10. 12

되었다.

건설적 귀납법(constructive induction)은 사례들이 갖고 있는 속성들에 적절한 연산자를 적용하여 이들 사례들을 좀 더 효율적으로 분류할 수 있는 새로운 속성들을 도출해 내는 기법이다 [5]. 건설적 귀납법은 기계학습 분야에서 개념 학습을 위한 학습 시스템의 구현 시 시스템의 학습 능력 향상을 위한 기법으로 활발한 연구가 이루어져 왔다. 본 연구의 목적은 주어진 사례의 집합으로부터 PROSPECTOR[6]에서 사용한 규칙 형태의 분류 규칙을 습득하는 유전 알고리즘 기반 귀납적 학습 환경에 적합한 건설적 귀납법을 개발하여 시스템의 학습 능력을 향상시키고자 하는 것이다.

본 논문의 구성은 다음과 같다. 2장에서는 분류 규칙의 습득을 위한 유전 알고리즘 기반 귀납적 학습 환경을 소개하고 3장에서는 유전 알고리즘 기반 귀납적 학습 환경에 적합한 건설적 귀납법을 제시한다. 4장에서는 건설적 귀납법이 유전 알고리즘 기반 학습 환경에 미치는 영향을 다양한 사례 집합을 이용하여 평가하고 5장에서는 본 논문의 결론을 제시한다.

Ⅱ. 유전 알고리즘을 이용한 분류 규칙의 습득

분류 규칙의 습득을 위한 유전 알고리즘 기반 학습 환경에서 훈련 사례 집합 내의 사례는 속성 $A_1, A_2, \cdots, A_n$에 대한 값 $a_1, a_2, \cdots, a_n$과 사례가 속한 클래스 C_k로 구성된 리스트, $(a_1, a_2, \cdots, a_n, C_k)$의 형태로 표현된다. 사례의 속성 A_i에 대한 값 a_i는 사례 집합 내에서 사례가 속성 A_i에 대해 갖고 있는 실제 속성 값보다 적은 값을 갖는 사례의 수를 사례의 실제 속성 값과 다른 값을 가지는 사례의 수로 나누어 0과 1사이의 값을 갖도록 정규화한 값이다. 이러한 정규화 과정은 사례가 규칙의 조건을 만족시키는 정도에 따라 규칙의 결론에 대한 가능성을 증가 혹은 감소시키면서 추론을 수행하는 PROSPECTOR에서 사용한 추론 방법의 적용을 가능하게 한다.

학습 시스템은 주어진 사례의 집합으로부터 "If E then C with S = s, N =n" 형태의 분류 규칙들을 습득한다. 여기서 S와 N은 사례가 규칙의 조건 E를 만족시키는 정도에 따라 규칙의 결론인 클래스 C에 속할 가능성을 증가 혹은 감소시키기 위해 제공되는 가능성에 대한 승수 값의 범위를 나타낸다. 습득된 규칙들로 구축된 분류 시스템에서 각각의 규칙들은 사례가 규칙의 조건을 완전하게 만족하면(즉, P(E') = 1) S의 값을, 불만족 시에는 (즉, P(E') = 0) N의 값을, 0 < P(E') < 1인 경우에는 P(E')의 값에 비례하여 N과 S사이의 값을 제공한다. 분류 시스템은 각각의 결론 C에 대한 확률 P(C)로부터 사례가 C에 속할 사전 가능성 O(C)를 식 O(C) = P(C)/(1 - P(C))에 따라 구하고 이에 C를 결론으로 갖는 규칙들이 제공하는 가능성에 대한 승수를 취합하여 C에 속할 사후 가능성 O(C')을 구한 후 사후 가능성이 가장 큰 클래스를 사례가 속한 클래스로 선택한다. 사례가 클래스 C에 속할 확률 P(C)는 사례 집합 내에서 C에 속하는 사례가 차지하는 비율로부터 구한다.

분류 시스템이 사례를 분류하는 과정은 다음과 같다:

1. 각각의 클래스 C_k에 대해 사례 집합에서 C_k에 속한 사례의 비율에 따라 $P(C_k)$를 구한 후 사전 가능성 $O(C_k) = P(C_k)/(1 - P(C_k))$를 구한다.

2. C_k를 결론절에서 참조하는 규칙
   (r_1) If E_1 then C_k with $S=s_1$, $N=n_1$
   ...
   (r_p) If E_p then C_k with $S=s_p$, $N=n_p$
   들이 제공하는 C_k에 대한 승수

$$\lambda_{ri} = \frac{O(C_k|E_i')}{O(C_k)} \qquad for\ i = 1, \ldots, p$$

   를 이용하여 사후 가능성

$$O(C_k') = O(C_k|E_1' \wedge \ldots \wedge E_p') = O(C_k) \times \Pi_{i=1}^{p} \lambda_{ri}$$

   을 구한다.

3. 사후 가능성이 가장 큰 클래스를 주어진 사례가 속한 클래스로 선택한다.

특정 속성에 대해 사례들이 갖는 값의 상대적인 대소관계나 속성 값들이 특정 값을 중심으로 하여 분포하는 성향 등은 서로 다른 클래스에 속한 사례들을 분류하는 기준으로 고려해 볼 수 있는 일반적 특징이라 할 수 있다. 이러한 직관적 고찰에 따라 학습 시스템은 주어진 사례 집합으로부터 두 가지 유형의 분류 규칙을 습득한다. 분류 규칙의 형태 중 하나는 "If is-high(A) then C with S = s, N = n"의 형태로 이 타입의 규칙은 고려대상이 되는 속성 A의 값의 상대적인 높고 낮음에 따라 사례가 C에 속할 가능성에 대한 승수를 식 λ = S*P(E') + N*(1 - P(E'))

for i = 1, ..., p

을 이용하여 N과 S사이의 값으로 제공한다 (그림 1-(a) 참조).

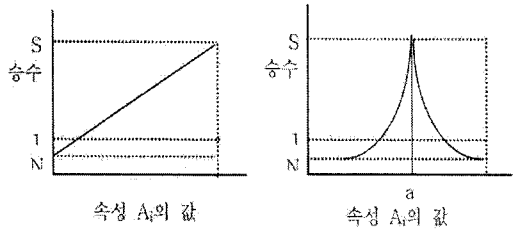

그림 1. 승수의 계산 (a) is-high 규칙 (b) is-close 규칙
Fig. 1 Computation of Odds-multiplier (a) is-high rule (b) is-close rule

다른 하나는 "If is-close(A, a) then C with S = s, N= n"의 형태로 고려하는 속성 A의 값이 어떤 특정 값 a에 근사한 정도에 따라 식 P(E') = P(is-close(A, a')= max(0, $1-2*(|a'-a|)^{1/2}$)을 이용하여 P(E')을 구한 후 클래스 C에 대해 N과 S사이의 값을 제공 한다 (그림 1-(b) 참조).

If is-high(A_4) then C_1 with S=100, N=0.016
If is-high(A_2) then C_2 with S=0.002, N=990
If is-close(A_1, 0.2) then C_2 with S=0.8, N=534
...

그림 2. 습득된 규칙 집합의 예
Fig. 2 Example of Learned Rule-set

그림 2는 습득된 규칙 집합의 예를 보인 것이다. 위의 규칙 집합에서 첫 번째 규칙은 A_4의 값이 클수록 사례가 C_1에 속할 가능성이 높은 것으로 간주하여 C_1의 가능성을 향상 시킬 수 있도록 S에 가까운 승수를 제공하고 A_4의 값이 작을수록 C_1일 가능성을 감소시키도록 N에 가까운 승수를 제공한다. 세 번째 규칙은 A_1의 값이 0.2에 가까울수록 사례가 C_2에 속할 가능성에 대해 S에 가까운 승수를 제공하여 그 가능성을 감소시키며 0.2에 멀어질수록 N에 가까운 승수를 제공하여 C_2의 가능성을 증가 시켜준다.

학습 시스템은 사례들을 분류하기 위해 필요한 속성들을 적절히 고려한 규칙들을 각각의 규칙에 필요한 s, n, 상수 a의 값과 함께 유전 알고리즘을 이용하여 습득한다. 유전 알고리즘을 이용한 학습 시스템의 구현에서 개체 집단은 일정 수의 규칙 집합으로 구성되며 각각의 규칙 집합은 임의의 수의 분류 규칙으로 구성된다. 초기의 개체 집단은 난수 발생기를 이용하여 임의의 수의 분류 규칙들로 이루어진 일정 수의 규칙 집합을 생성함으로써 얻어진다. 계속되는 진화 과정에서는 적합도에 비례하여 선택된 규칙 집합에 대해 교배 연산자와 돌연변이 연산자 등의 유전 연산자를 적용하여 새로운 개체 집단을 생성하는 과정을 원하는 해가 얻어질 때까지 반복한다. 적합도는 규칙 집합이 사례들을 어느 정도 정확하게 분류 할 수 있는가 하는 분류의 정확도를 이용하여 평가한다.

III. 건설적 귀납법의 구현

2장에서 설명한 학습 시스템이 습득하는 규칙은 조건절에서 하나의 속성만을 고려하도록 되어 있는데 이러한 제한된 형태의 규칙은 속성들 사이에 존재하는 연관성으로부터 도출될 수 있는 유용한 정보를 습득하는데 효율적이지 못하여 일부 분류 문제에 대해 학습 시스템의 성능을 저하시키는 주요한 요인이 된다.

이러한 문제점을 해결하여 학습 시스템의 성능을 향상시키는 한 가지 방법은 규칙의 조건절에 논리 연산자를 도입하여 표현력을 향상시키는 것이다. 즉 AND, OR, NOT등의 논리 연산자를 도입하여 조건절에서 다수의 속성을 고려하여 특정 클래스에 대한 가능성의 승수를 제공하도록 규칙의 형태를 개선하는 것이다. 그러나 이 방법은 유전 알고리즘이 규칙의 습득 과정에서 다루어야 할 탐색 공간을 크게 확장 시키는 결과를 낳게 되는 문제점과 함께 무분별한 논리 연산자의 사용으로 인해 의미 없는 규칙을 생성하거나 혹은 일반성을 잃어버리고 특수한 상황이나 사례에 적합한 규칙을 생성하게 하는 문제점을 갖게 된다.

학습 시스템의 성능을 향상시키기 위해 취할 수 있는 다른 한 가지 방법은 건설적 귀납법을 적용하는 것이다. 건설적 귀납법의 사용 예로 속성 A_1, A_2에 대한 속성 값과 사례가 속한 클래스(C_1 또는 C_2)로 나타내진 4개의 사례 $e_1 = (0, 0, C_1)$, $e_2 = (1, 0, C_2)$, $e_3 = (0, 1, C_2)$, $e_4 = (1, 1, C_1)$에 대해 적절한 분류 규칙을 습득하는 문제를 생각해 보자. 이 분류 문제는 사실 전형적인 XOR-문제이다. 2장에서 언급한 학습 환경 하에서는 규칙이 조건절에서 하

나의 속성을 고려하도록 한 제약 때문에 주어진 XOR-문제를 해결하기 위한 적절한 규칙 집합을 쉽게 습득하지 못하는 경향이 있으나 속성 A_1, A_2로부터 XOR연산자를 적용하여 유도된 새로운 속성 $A_3 = A_1$ XOR A_2를 추가하여 얻어진 사례 $e_1' = (0, 0, 0, C_1)$, $e_2' = (1, 0, 1, C_2)$, $e_3' = (0, 1, 1, C_2)$, $e_4' = (1, 1, 0, C_1)$에 대해서는 학습 시스템은 조건절에서 A_3를 고려한 규칙을 습득하여 사례들을 올바르게 분류하게 된다. 위의 예제는 간단하지만 건설적 귀납법의 유용성을 보여주는 적절한 예이다.

만일 건설적 귀납법을 통해 얻어진 새로운 속성 중 분류에 유용한 속성들을 선별하여 추가할 수 만 있다면 건설적 귀납법을 적용하는 방법은 유전 알고리즘의 성능에 큰 영향을 미치지 않도록 탐색 공간을 효율적으로 제어하면서 앞 서 언급한 논리 연산자를 도입하는 방법과 유사한 효과를 얻을 수 있으며, 2장에서 설명한 기존의 학습 환경을 그대로 사용할 수 있다는 이점이 있다. 그러나 건설적 귀납법을 적용하려면 주어진 속성들로부터 새로운 속성을 유도해내기 위한 기법과 함께 유도된 속성이 사례들을 분류하는데 유용한지를 평가하기 위한 기법 등이 개발되어져야 한다.

본 연구에서는 건설적 귀납법의 적용을 위해 기존의 속성으로부터 새로운 속성을 유도하기 위한 속성 결합 연산자를 다음과 같이 정의하였다.

- $\otimes$ 연산자 : $val(e, A \otimes B) = val(e, A) * val(e, B)$
- $\oplus$ 연산자 : $val(e, A \oplus B) = val(e, A) + val(e, B) - val(e, A) * val(e, B)$
- $\odot$ 연산자 : $val(e, \odot A) = 1 - val(e, A)$

위의 식에서 $val(e, X)$는 사례 e가 속성 X에 대해 갖는 속성 값을 의미한다. 새로운 속성은 주어진 속성의 집합에서 임의의 속성들을 선택한 후 속성 결합 연산자를 적용하여 위에서 제시한 방법에 따라 각각의 사례에 대해 속성 값을 계산함으로써 얻어진다.

주어진 속성으로부터 새로운 속성을 유도해 내고 나면 새로운 속성이 분류 작업에 유용한 속성인가를 판단해 낼 수 있는 기법이 필요하다. 학습 환경에서 습득하는 규칙은 is-high와 is-close의 형태이므로 두 규칙의 형태에 따라 각각 다른 기준이 필요하다.

Is-high규칙은 사례가 갖고 있는 속성 값의 상대적 크기를 고려하여 사례가 특정 클래스에 속할 가능성에 대한 승수를 제공하므로 어떤 속성에 대해 다른 클래스에 속한 사례들이 특정 값을 경계로 하여 서로 반대 방향으로 편향되게 분포 한다면 이 속성은 이러한 성향을 갖지 않는 다른 속성에 비해 is-high규칙을 이용한 분류 작업에 더 적합한 속성이라 할 수 있다. 이러한 직관적 고찰에 준하여 유도된 속성이 is-high 규칙에 적합한 속성인지를 판단할 수 있는 평가 기준을 다음과 같이 개발하였다.

우선 훈련 사례 집합 내의 사례 중 임의의 두 클래스 C_i와 C_j에 속한 사례들이 속성 X에 대해 갖고 있는 값들의 집합을 $VS(X, C_i, C_j) = \{v_1, \cdots, v_n\}$라 하고 $(v_{i-1} < v_i < v_{i+1})$, $v_k \in VS(X, C_i, C_j)$인 임의의 값 v_k에 대해 $NE(X, v_k, C_i)$를 클래스 C_i에 속한 사례 중 속성 X에 대해 값 v_k를 갖는 사례의 수를 반환하는 함수로 정의하자. 그리고 함수 $MIS(X, C_i, C_j, v_k)$와 $MMIS(X, C_i, C_j)$를 다음과 같이 정의하자.

$$MIS(X, C_i, C_j, v_k) = \min \{ \sum_{m=k}^{n} NE(X, v_m, C_i) + \sum_{m=1}^{k-1} NE(X, v_m, C_j), \sum_{m=k}^{n} NE(X, v_m, C_j) + \sum_{m=1}^{k-1} NE(X, v_m, C_i) \}$$

$$MMIS(X, C_i, C_j) = \min_{v_k} \{ MIS(X, C_i, C_j, v_k) \}$$

여기서 mim은 최소값을 반환하는 함수이다. 이들 정의에 대한 예로 앞에서 언급한 XOR-문제를 고려해 보면 $VS(A_1, C_1, C_2) = \{0, 1\}$, $NE(A_1, 0, C_1) = 1$, $NE(A_1, 1, C_1) = 1$, $MIS(A_1, C_1, C_2, 0) = 2$, $MMIS(A_1, C_1, C_2) = 2$로 주어진다. 이 정의에서 $MMIS(X, C_i, C_j)$는 속성 X에 대해 어떤 값을 기준으로 양분하여 C_i와 C_j에 속한 사례들을 분류하는 경우 잘 못 분류되는 사례 수의 최소값을 나타낸다.

위에서 정의한 MMIS 함수를 이용하여 속성 A, B로부터 속성 결합 연산자를 적용하여 유도된 속성 D가 is-high 형태의 규칙에 유용한가를 다음의 식을 이용하여 평가한다.

$$MMIS(D, C_i, C_j) \leq \alpha * \min \{MMIS(A, C_i, C_j), MMIS(B, C_i, C_j)\} \quad (1)$$

식 (1)에서 α 는 0 과 1사이의 실수 값을 갖는다. 새로 유도된 속성이 식 (1)을 만족한다면 새로운 속성에 대한 MMIS의 값이 속성을 유도하는데 사용된 속성들에 비해 $(1 - \alpha)$만큼 감소한 값을 갖게 됨을 의미한다. 이는 결

국 C_i와 C_j에 속한 사례들이 기존의 속성에 비해 새로 유도된 속성에 대해 어떤 값을 기준으로 서로 반대 방향으로 좀 더 편향되게 분포하는 경향이 있음을 의미하며, 따라서 새로 유도된 속성은 C_i와 C_j에 속한 사례들의 분류 작업 시 is-high 규칙의 속성으로 유용한 것으로 볼 수 있다.

앞서 언급한 XOR-문제의 경우에 $MMIS(A_3, C_1, C_2) = 0$, $MMIS(A_1, C_1, C_2) = 2$, $MMIS(A_2, C_1, C_2) = 2$이 되므로 α 를 0.9로 하면 (즉 10% 감소하면 유용한 속성으로 간주함) A_3는 A_1, A_2에 비해 is-high규칙을 이용한 분류 작업에 더 적합한 속성으로 볼 수 있다.

Is-close 규칙은 사례가 갖고 있는 값이 특정 값에 근접한 정도에 따라 결론절에서 참조하는 클래스에 대해 승수를 제공한다. 따라서 어떤 속성에 대해 특정 값들을 중심으로 같은 클래스에 속한 사례들끼리 모이는 성향이 있다면 이 속성은 이러한 성향을 갖지 않는 다른 속성에 비해 is-close 규칙에 더 적합한 속성이라 할 수 있다. 이러한 직관적 고찰에 준하여 유도된 속성이 is-close 규칙에 적합한 속성인지를 판단할 수 있는 평가식을 다음과 같이 개발하였다.

우선 앞서 정의한 VS, NE의 개념에 더하여 집합 VS 내의 각 속성 값 v_k에 대해 $NE(X, v_k, C_i)$와 $NE(X, v_k, C_j)$의 값을 비교하여 큰 값을 갖는 클래스에 값 v_k가 속한다고 정의하자. 이 정의에 대한 예로 앞에서 언급한 XOR-문제를 고려해 보면 $VS(A_3, C_1, C_2) = \{0, 1\}$이 되고 속성 값 0는 C_1에, 1은 C_2에 속하게 된다. 이제 VS내의 인접한 속성 값들 중 같은 클래스에 속한 것들을 한데 묶어 이들 속성 값으로 이루어진 집합을 정의하면 VS로부터 속성 값들의 부분 집합들로 구성 된 집합 $VSS(X, C_i, C_j) = \{SV_1, \cdots, SV_p\}$을 얻을 수 있다. 여기서 SV_i는 특정 클래스에 속한 인접한 속성 값들의 집합을 의미한다. 예를 들어 앞에서 언급한 XOR-문제를 고려해 보면 $VSS(A_3) = \{\{0\}, \{1\}\}$이 된다. 이제 $SV_k \in VSS(X, C_i, C_j)$인 SV_k에 대해 함수 NESV와 MCC를 다음과 같이 정의 하자.

$$NESV(X, SV_k, C_i) = \sum_{v \in SVk} NE(X,v,C_i)$$

$$MCC(X, C_i, C_j) = \sum_{k=1}^{p}(\max\{NESV(X,SV_k,C_i,), NESV(X,SV_k,C_j)\} - \min\{NESV(X,SV_k,C_i), NESV(X,SV_k,C_j)\}$$

위 정의에서 함수 $NESV(X, SV_k, C_i)$는 C_i에 속한 사례 중 X에 대한 속성 값으로 SV_k에 속한 값을 갖는 사례의 수를 반환하는 함수이고 $MCC(X, C_i, C_j)$는 VSS내의 모든 원소에 대해 $NESV(X,SV_k,C_i)$와 $NESV(X,SV_k,C_j)$의 값 중 큰 값과 작은 값의 차를 구해 합한 것으로 C_i와 C_j에 속한 사례들이 속성 X에 대해 어떤 값들을 중심으로 동 종류의 사례들끼리 분포하는 성향이 더 많을수록 MCC는 큰 값을 갖게 된다. 이렇게 정의 된 함수를 이용하여 속성 A, B로부터 유도된 속성 D가 is-close 형태의 규칙에 유용한가를 다음의 식을 이용하여 평가한다.

$$MCC(D,C_i,C_j) \geq \beta * \max\{MCC(A,C_i,C_j), MCC(B,C_i,C_j)\} \quad (2)$$

식 (2)에서 β 는 1보다 큰 실수 값을 갖는다. 식 (2)의 의미는 유도된 속성 D에 대한 MCC의 값이 이 속성을 유도하기 위해 사용된 속성 A, B에 비해 β 만큼 큰 값을 갖는 다면 이는 C_i와 C_j에 속한 사례들이 속성 D에 대해 어떤 값들을 중심으로 동 종류의 사례들끼리 분포하는 성향을 좀 더 많이 갖고 있다는 것을 의미하므로 유도된 새로운 속성은 C_i와 C_j에 속한 사례들의 분류 작업 시 is-close 규칙에 유용한 것으로 간주할 수 있다는 것이다.

앞서 언급한 XOR-문제에서 속성 A_1, A_2, A_3에 대해 VS, VSS, MCC을 구해 보면 다음과 같다.

$VS(A_1) = VS(A_2) = VS(A_3) = \{0, 1\}$,
$VSS(A_1) = VSS(A_2) = \varnothing$, $VSS(A_3) = \{\{0\}, \{1\}\}$
$MCC(A_1,C_1,C_2) = MCC(A_2,C_1,C_2) = 0$
$MCC(A_3, C_1, C_2) = 4$

β 를 1.1로 정하면 A_1, A_2, A_3는 식 (2)를 만족하므로 A_3는 A_1, A_2에 비해 is-close규칙을 이용한 분류 작업에 더 적합한 속성으로 볼 수 있다.

Ⅳ. 건설적 귀납법이 학습 환경에 미치는 영향 평가

건설적 귀납법이 학습 시스템의 성능에 미치는 영향을 평가하기 위해 다음의 사례 집합을 이용하였다 (이들 사례 집합은 "UCI machine learning repository" 에서 습득하였음)

- 붓꽃 사례 집합: 3가지 종류의 붓꽃으로부터 얻어진 150개의 사례가 꽃잎의 길이와 넓이, 꽃받침의 길이와 넓이의 4가지 속성 값과 붓꽃의 종류를 나타내는 값으로 표현된 사례 집합
- 레이더 시그널 사례 집합: 올바른 경우와 잘못된 경우의 351개 레이더 시그널 사례가 34가지의 속성 값과 사례가 속한 클래스로 표현 됨
- 당뇨 환자 사례 집합: 당뇨 환자인 경우와 정상인인 경우로 분류되는 768개의 사례가 8가지 속성 값과 사례가 속한 클래스로 표현됨
- 콩의 질병 사례 집합: 콩에 감염될 수 있는 15가지 질병으로부터 얻어진 290개의 사례가 35개의 속성 값과 사례가 속한 클래스로 표현됨

우선 기존의 학습 시스템의 성능을 평가하기 위해 사례 집합을 크기가 같은 두 개의 부분 집합인 훈련 사례 집합과 평가 사례 집합으로 나누어 훈련 사례 집합을 이용하여 분류 시스템을 구축한 후 평가 사례 집합을 이용하여 분류 시스템의 성능을 평가하는 과정을 5회 반복하였다. 건설적 귀납법이 학습 환경에 미치는 영향을 평가하기 위해서는 기존의 사례 집합에 대해 건설적 귀납법을 적용하여 새로운 속성들을 추가한 사례 집합을 구한 후 훈련 사례 집합을 이용하여 분류 시스템을 구축하고 평가 사례 집합으로 분류 시스템의 성능을 평가하는 과정을 5회 반복하였다.

건설적 귀납법은 다음의 과정을 거쳐 적용하였다. 사례 집합이 긍정적 사례와 부정적 사례로 구성된 경우에는 우선 사례 집합 내의 각각의 속성에 '⊙' 연산자를 적용하여 기존의 속성들과 '⊙' 연산자를 적용하여 얻어진 속성들의 집합을 구한다. 그런 다음 이 속성 집합에서 두 개의 속성을 선택하여 얻어지는 모든 속성의 쌍에 대해 '⊗'와 '⊕' 연산자를 적용하여 새로운 속성을 유도해 낸 후 유용한 것으로 평가된 경우에 이를 기존의 속성에 추가하였다. 이 과정에서 '⊙' 연산자를 적용하여 습득된 속성들은 다음 단계에서 '⊗'와 '⊕' 연산자를 적용하여 새로운 속성을 유도하는 과정에만 사용되고 기존의 속성에 유용한 속성으로 추가되지는 않는데 그 이유는 2장에서 논한 규칙들에서 N과 S값을 $N \geq 1, S < 1$이 되도록 하면 그 속성에 대해 '⊙'을 적용한 것과 같은 의미를 갖는 규칙을 습득할 수 있기 때문이다. 이러한 과정을 거쳐 기존의 속성과 새로 추가된 속성으로 사례 집합을 만든 후 이 사례 집합에 대해 건설적 귀납법을 1회 더 적용하였다. 유도된 속성의 유용성을 평가하는 평가식에서 α 와 β 는 평가 기준의 수위를 조절해 주는 역할을 한다. α (β)의 값이 클(작을)수록 평가 기준은 약화되어 더 많은 속성들이 분류에 유용한 것으로 판정되어 사례 집합에 포함되고 반대로 α (β)값이 작을(클)수록 평가 기준이 강화되어 보다 적은 수의 속성들이 유용한 속성으로 판정되게 된다. 본 실험에서는 α 와 β 값을 각 각 0.9와 1.1을 기준으로 하여 적정수의 속성이 추가되도록 사례 집합에 따라 값을 증감하여 사용하였다. 여러 개의 클래스로 구성된 사례 집합에 대해서는 각각의 클래스에 대해 그 클래스에 속한 사례들을 긍정적 사례로 나머지 다른 클래스에 속한 사례들을 부정적 사례로 간주하여 앞서 설명한 방법에 따라 건설적 귀납법을 적용하였다. 이러한 과정을 거쳐 건설적 귀납법을 적용할 경우 붓꽃, 레이더 시그널, 당뇨 환자, 콩의 질병 사례 집합에 대해 각각 4, 10, 4, 20개의 새로운 속성이 추가되었다.

표 1은 학습 시스템의 성능을 비교하여 보인 것이다. 표에서 BASIC은 기존의 학습 시스템을 의미하고 CI는 건설적 귀납법을 적용한 경우를 나타낸다. 표 1은 건설적 귀납법을 적용한 경우에 레이더 시그널의 훈련 사례 집합에 대해서는 사례의 95.5%를, 평가 사례 집합에 대해서는 사례의 87.9%를 올바르게 분류함으로써 건설적 귀납법을 적용하지 않은 경우에 비해 훈련 사례에 대해서는 1.9%, 평가 사례에 대해서는 2.0% 분류의 정확도를 향상시키고 있음을 보인다.

표 1. 분류 성능의 비교
Table. 1 Comparison of Classification Performance

사례집합	BASIC		CI	
	훈련	평가	훈련	평가
붓꽃	98.3	94.0	98.9	94.3
레이더	93.6	85.9	95.5	87.9
당뇨	78.2	74.4	79.0	74.7
콩의질병	55.5	47.1	60.5	51.7

그림 3은 학습 시스템이 습득한 최적의 분류 규칙 집합의 분류의 정확도를 세대별로 보인 것이다. 그림 3은 건설적 귀납법을 적용한 경우 기존의 속성에 새로운 속성이 추가되어 탐색 공간이 확대되었음에도 불구하고 학습 시스템의 성능은 오히려 향상되었음을 보인다. 특

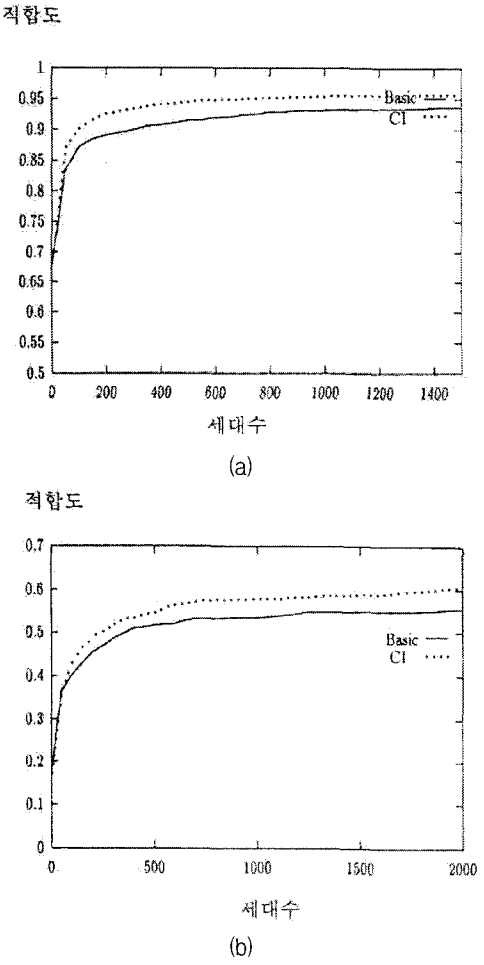

그림 3. 최적 분류 규칙 집합의 적합도 비교
(a) 레이더 시그널 (b) 콩의 질병
Fig. 3 Comparison of Best Rule-set's Fitness
(a) Radar Signal (b) Soybean Diseases

히 콩의 질병 사례의 경우 추가된 속성이 20개나 되는데도 불구하고 학습 시스템은 좀 더 효율적으로 분류 규칙 집합을 습득하고 있음을 보인다.

분류 시스템의 성능을 향상시키는 기법 중 하나는 주어진 사례 집합에 대해 다수의 분류기를 습득한 후 이를 통합하여 분류 시스템을 구축하는 것이다 [7]. 2장에서 설명한 유전 알고리즘 기반 학습 환경 하에서는 난수 발생기에 의존하는 탐색과정으로 인해 난수 발생기에 사용하는 초기 값을 달리 하면 학습 시스템은 다른 탐색 공간을 탐색하게 되어 결과적으로 다른 학습 결과, 즉 분류 규칙 집합을 습득하게 된다. 따라서 이러한 유전 알고리즘 기반 학습 환경의 특성을 이용하면 주어진 사례 집합에 대해 학습 시스템을 반복 실행하여 다수의 분류 규칙 집합을 습득할 수 있다. 표 2는 다수의 분류 규칙 집합을 습득한 후 이를 통합하여 구축한 분류 시스템의 성능을 보인 것이다.

표 2. 다중 분류기 시스템의 성능
Table. 2 Performance of Multi-classifier System

	붓꽃	레이더	당뇨	콩의질병
훈련사례	100	98.2	82.0	75.9
평가사례	96.0	92.3	76.3	68.4

표 2는 다수의 분류기를 이용하여 구축한 분류 시스템이 붓꽃 사례 집합에 대해 훈련 사례 집합의 경우 100%, 평가 사례 집합의 경우 96%의 정확도로 사례를 분류하고 있음을 보인다.

본 연구에서 구축한 분류 시스템의 성능을 신경망과 결정 트리 알고리즘을 이용하여 구축된 분류 시스템의 성능과 비교하였다. 표 3에서 '유전'은 본 논문에서 제시한 유전 알고리즘 기반 학습 환경 하에서 다수의 분류기를 이용하여 구축한 분류 시스템의 성능을 나타내고 '신경망'과 'C4.5'는 각각 신경망과 C4.5 결정 트리 알고리즘을 이용하여 구축한 분류 시스템의 성능을 보인 것이다.

표 3. 학습 시스템의 성능 비교
Table. 3 Comparison of Learning Systems

사례 집합	유전	신경망	C4.5
붓 꽃	96.0	95.9	95.9
당 뇨	76.3	76.8	72.4

표 3은 본 논문에서 제시한 학습 환경 하에서 습득된 분류 시스템이 평균적으로 붓꽃 사례의 96%를 올바르게 분류한 반면 신경망과 C4.5는 95.9%를 올바르게 분류하여 본 연구를 통해 구축한 학습 시스템이 신경망과 결정 트리 알고리즘에 견줄만한 학습 능력이 있음을 보인다.

V. 결 론

본 논문에서는 주어진 사례의 집합으로부터 이들 사례들을 분류할 수 있는 규칙들을 습득하는 유전 알고리즘 기반 귀납적 학습 환경에 적합한 건설적 귀납법을 제시하였다. 다양한 사례 집합을 이용한 실험 결과는 본 논문을 통해 제시된 건설적 귀납법이 학습 시스템의 학습 능력을 향상시키고 있음을 보인다. 주어진 속성으로부터 유도된 새로운 속성이 분류 작업에 유용한 속성인가를 평가하기 위해 개발된 속성 평가 기법은 일반적인 평가 기법으로 사용 가능하며 따라서 본 연구를 통해 구현된 건설적 귀납법은 유전 알고리즘 기반 학습 환경 외에 신경망과 같은 다른 계산 메커니즘에 기반을 둔 학습 시스템의 성능 향상을 위한 기법으로도 활용 가능하리라 사료된다.

참고문헌

[1] M. Srinivas and L. M. Parnaik, "Genetic algorithms: a survey," *IEEE Computer*, Vol. 27, pp. 17-26, June 1994.

[2] G. Roberts, "Dynamic planning for classifier systems," in *Proc. 5th Int. Conf. Genetic Algorithms*, pp. 231-237, 1993.

[3] C. K. Chiang, H. Y. Chung, and J. J. Lin, "A self-learning fuzzy logic controller using genetic algorithms with reinforcements," *IEEE Trans. Fuzzy Systems*, Vol. 5, pp. 460-467, 1997.

[4] H. Ishibuchi and T. Nakashima, "Improving the Performance of Fuzzy Classifier Systems for Pattern Classification Problems with Continuous Attributes," *IEEE Transactions on industrial electronics*, Vol. 46, No. 6, pp. 1057 - 1068, December 1999.

[5] S. Markovitch and D. Rosenstein, "Feature Generation Using General Constructor Functions," *Machine Learning*, 49, pp. 59-98, 2002.

[6] R. Duda, P. Hart and J. Nilsson, "Subjective Bayesian methods for rule-based inference systems,' in *Proc. National Computer Conference*, pp. 1075 - 1082, 1976

[7] R. Meir and G. Rätsch, "An introduction to boosting and leveraging," *Advanced Lectures on Machine Learning*, pp. 119-184, 2003.

저자소개

김 영 준(Yeongjoon Kim)

1984년 고려대학교 산업공학과 (공학사)
1996년 미국 Univ. of Houston 전자계산학과(박사)
1997년 ~ 현재 상명대학교 소프트웨어학부 부교수
※관심분야 : 기계학습, 진화알고리즘, 전문가시스템