한국 퍼지 및 지능시스템학회 논문지 1999, Vol. 9, No. 4. (1999)367~374

개념 상승과 속성의 최적 감축에 의한 결정 규칙의 생성

Generation of Decision Rules Based on Concept Ascension and Optimal Reduction of Attributes

정 홍 · 정환묵*

Hong Chung and Hwan Mook Chung*

계명대학교 컴퓨터전자공학부 , * 대구효성가톨릭대학교 전자정보공학부

요 약

본 논문은 대규모 데이터베이스에서 의사결정을 위한 지식을 효율적으로 추출하기 위해 개념 상승과 속성 감축에 기반한 통합적 방법을 제안한다. 본 방법은 클러스터링 기법에 의해 개념 트리를 자동 생성하고, 개념 상승 기법에 의해 데이터베이스를 일반화하며, 속성의 중요도를 사용한 속성 감축에 의해 최적 감축을 하고, 식별가능 행렬과 함수를 사용하여 효율적으로 속성값을 감축하여 최적의 최소 결정 규칙을 유도한다. 본 방법은 투자 계획이나 가격 결정과 같은 의사결정 업무, 각종 고장 진단이나 의료 진단을 위한 지식베이스의 구축, 마케팅 분석이나 실험 데이터 분석, 고수준의 질의에 의한 정보검색 등에 효과적으로 사용될 수 있다.

ABSTRACT

This paper suggests an integrated method based on concept ascension and attribute reduction for efficient induction of decision rules from a large database. We study an automatic scheme to generate concept trees by a clustering technique, a method for generalizing databases by the concept ascension technique, an optimal reduction method by means of attributes reduction using the significance of attributes, and an efficient way of reduction of attribute values applying the discernible matrix and functions. The method can be used for the decision making tasks such as an investment planning or price evaluation, the construction of knowledge bases for diagnosis of defects or medical diagnosis, data analysis such as marketing or experimental data, information retrieval for high level inquiries, and so on.

1. 서 론

데이터베이스에서 유용한 지식을 추출하기 위한 방법인 데이터 마이닝(Data Mining) 또는 KDD (Knowledge Discovery in Databases)가 데이터베이스나 기계학습 연구자들에게 중요한 연구과제로 대두되고 있다[5]. 데이터 마이닝이란 데이터에 내재되어 있는 정보와 지식을 정교한 분석모형을 사용하여 검색하는 작업으로서, 기존의 지식을 재확인하는 역할을 수행함과 동시에 지금까지 인식하지 못했던 새로운 정보와 지식을 제공한다[22].

데이터베이스에서 지식발견의 핵심은 특정 형태로 저장된 정보를 일반화된 문장이나 특성에 대한 규칙, 혹은 데이터 내의 관계로 변화시키는 것이다[18]. 이를 위한 방법에는 Han[7]의 속성중심 귀납법, Quinlan [14]이 제안한 정보 이론에 기초한 결정 트리에 의한 귀납법, Pawlak[13]의 라프셋(rough sets) 이론에 의한 지식감축[12] 방법 등이 있다. 이 방법들은 각기 특징을 갖고 있으나 다음과 같은 한계점도 있다[8,16]. 즉, 속성중심 귀납법은 데이터베이스에 개념 상승에 의한 일반화를 적용함으로써 추상성이 높은 특성, 분류, 연관 등의 규칙을 생성하나, 속성간에 종속관계를 분석하지 않고 규칙으로 변환함으로써 규칙이 중복 정보와 불필요한 제약을 포함할 가능성이 있어 유도된 규칙이 간략성이 없고 복잡하다. 결정트리에 의한 귀납법은 속성의 종속관계를 엔트로피(entropy)라는 정보값을 이용하여 트리 형태의 그래프로 나타냄으로써 분류 규칙을 간단하게 유도하나, 속성 수가 많은 데이터베이스에 적용할 때는 처리시간이 급격하게 많아지는 효율성 문제와 이산적(discrete)이지 못한 데이터에는 적용할 수 없는 점이 있고, 수치 데이터에 대해서는 개념적 규칙으로 표현하기 어려운 한계가 있다[10]. 그리고 라프셋 이론에 의한 감축 방법은 속성집합을 수학적으로 엄격하게 분석함으로써 불필요한 속성과 속성값을 정확하게 제거하여 최소 규칙을 유도하나, 감축 가능성을 판단하는데 있어서 처리시간이 많이 소요되기 때문에 속성 및 튜플의 수가 많은 대형 데이터베이스에는 적용하기 어렵고 또한 그룹화가 안된 연

속적 데이터를 다룰 수 없는 한계가 있다.

상기와 같은 한계점을 해결하기 위한 방법이 일부 연구되고 있으나[1,8,20] 그 가능성 정도만 제시되었을 뿐 구현을 위한 구체적인 방안이 부족하다.

본 논문에서는 데이터베이스의 일반화를 위한 속성 중심 귀납법에서 사용하는 개념상승 기법과 라프셋에 의한 지식감축 방법을 복합하여 저수준의 데이터를 고수준 정보로 일반화하고, 불필요한 속성 및 속성값들을 감축하여 간략화된 결정규칙을 도출하는 통합방법의 데이터 마이닝 방법을 연구한다. 즉, 속성중심의 개념 상승에 의해 데이터를 일반화하고, 일반화된 속성간에 데이터의 종속성을 찾고 일반화된 관계를 표현하는 최소 속성집합을 선정하기 위해 라프셋의 속성 감축 기법을 적용한다. 이를 위해 개념상승에 사용할 개념트리(concept tree)를 자동 생성하고, 속성 감축을 효율적으로 하기 위해 속성의 중요도 계산에 결정트리 귀납법의 정보획득량 측정방법을 이용한다.

2. 개념 계층과 개념 상승

2.1 개념 계층

개념 계층은 데이터베이스의 속성에 있어서 일반화 관계의 집합이다. 일반화 관계는 속성값의 전체집합과 이를 일반화한 단일값간의 관계이다. 즉, 속성 a의 일반화 관계는 a의 정의역이 $\{A_1, A_2, \ldots A_k\}$이고 개념으로 표현된 단일값이 B일 때, $\{A_1, A_2, \ldots A_k\} \subset B$로 표현되며, 이때 B는 각 $A_i(1 \le i \le k)$의 일반화이다. 예를 들어 나이에 대한 일반화 관계는 다음과 같다.

0 ~ 5 ⊂ 유아
6 ~ 12 ⊂ 어린이
{유아, 어린이} ⊂ 소아
{10대, 20대, 30대} ⊂ 청년

속성에 대한 개념 계층은 낮은 수준의 속성값을 하위 계층에 구성하고, 개념의 수준을 높일수록 추상화된 개념으로 표현하기 위해 트리 구조로 구성하며, 노드간 링크(link)를 리프노드(leaf node)에서 루트노드(root node)로 향하게 하여 개념 트리를 구성한다[6,17].

개념 계층은 자동적으로 또는 반자동적으로 구성할 수 있는데[7], 수치 속성은 클러스터링 방법 혹은 통계적 방법에 의하여 이산적 개념 계층으로 자동 조직화될 수 있고, 비수치 속성은 속성간 유사도나 거리 등 상관 관계[3]에 의하여 구성될 수도 있으나 비실용적이므로 일반적으로 전문가의 지식을 이용한다. 본 논문에서 수치 속성에 대해서는 완전 자동화가 가능한 클러스터링 방법을 사용하고, 비수치 속성에서는 실용적이고 간단한 전문가 지식을 이용하고자 한다.

수치 속성은 Fisher가 제안한 개념적 클러스터링 시스템인 COBWEB[4]에 의하여 자동으로 조직화할 수 있는데, 이는 속성집합으로 기술된 객체를 분류 트리로 구성하는데 CU(Category Utility)라는 품질 척도를 사용한다. 즉, 클러스터 C를 n개의 상호배타적 클래스 $C_1, \ldots, C_n$으로 분할하는데 있어서 CU는 분할 후 클래스내의 유사성(intra-class similarity) 및 클래스간의 상이성(inter-class dissimilarity)을 의미하는 적합도(goodness)의 증가로 정의한다. 이 방법은 분류하는데 많은 메모리와 시간을 소요하므로 범주 데이터에만 적용되고 수치 데이터에는 사용하기 어렵다[3]. 본 논문에서는 Chu 등이 개발한 CoBase[3]에서 지식베이스를 구축하기 위한 TAH(Type Abstraction Hierarchies)의 생성에 CU를 근사적으로 계산하는 방법을 속성 단위 및 이진 분할 단위로 간략화하여 개념 트리의 자동 생성에 사용한다. TAH에서는 클러스터링의 척도로서 RE(Relaxation Error)를 사용하는데, 클러스터 C가 x_i의 집합으로 구성되어 있을 때 실제 속성값과 일반화한 값간의 평균 차이로 정의한다.

속성값 x_i의 $RE(x_i) = \sum_{j=1}^{n} P(x_j)|x_i - x_j|$
$P(x_j)$: C에서 속성값 x_j의 발생 확률

$RE(x_i)$를 C의 모든 속성값 x_i에 대하여 합하면 다음과 같다.

C 전체의 $RE(C) = \sum_{i=1}^{n} P(x_i)RE(x_i)$

C의 분할 $P=\{C_1,\ldots C_n\}$에서 분할 P의 RE는 다음과 같이 정의한다.

$RE(P) = \sum_{k=1}^{n} P(C_k)RE(C_k)$
$P(C_k)$: C_k의 속성값 수를 C의 속성값 수로 나눈 값

일반적으로 RE(P)<RE(C)인데, 이는 분할함으로써 RE가 감소함을 의미하므로, 최적 분할은 가장 적은 값을 갖는 RE(P)를 갖도록 분할한다.

예를 들어 C = {1, 2, 2, 3, 3, 3, 4, 4, 5, 5}와 같은 클러스터가 있을 때, RE(C)는 다음과 같이 계산된다.

$RE(1) = (1/10)\times 0 + (2/10)\times 1 + (3/10)\times 2 + (2/10)\times 3 + (2/10)\times 4 = 2.2$
$RE(2) = (1/10)\times 1 + (2/10)\times 0 + (3/10)\times 1 + (2/10)\times 2 + (2/10)\times 3 = 1.4$
$RE(3) = 1,\ RE(4) = 1.2,\ RE(5) = 1.8$
$RE(C) = (1/10)\times 2.2 + (2/10)\times 1.4 + (3/10)\times 1 + (2/10)\times 1.2 + (2/10)\times 1.8 = 1.4$

여기서 클러스터 C를 2등분 한다고 할 때 4가지 경

우가 있으며, 각 분할의 RE는 다음과 같이 계산된다.

$P_1 = \{C_1, C_2\} = \{\{1\}, \{2, 2, 3, 3, 3, 4, 4, 5, 5\}\}$
$RE(C_1) = 0$
$RE(C_2) = 1.2$
$RE(P_1) = (1/10)\times 0 + (9/10)\times 1.2 = 1.08$
$P_2 = \{C_1, C_2\} = \{\{1, 2, 2\}, \{3, 3, 3, 4, 4, 5, 5\}\}$
$RE(C_1) = 0.44$
$RE(C_2) = 0.9$
$RE(P_2) = (3/10)\times 0.44 + (7/10)\times 0.9 = 0.76$
$P_3 = \{C_1, C_2\} = \{\{1, 2, 2, 3, 3, 3\}, \{4, 4, 5, 5\}\}$
$RE(P_3) = 0.67$
$P_4 = \{C_1, C_2\} = \{\{1, 2, 2, 3, 3, 3, 4, 4\}, \{5, 5\}\}$
$RE(P_4) = 0.85$

위의 4가지 분할중 P_3로 분할하는 것이 RE가 가장 적다.

그런데 하나의 클러스터를 t개의 서브클러스터로 분할하는 조합의 수는 t에 지수적이므로 최적분할 계산은 지수적 시간복잡도를 가진다. 따라서 본 논문에서는 계산시간을 줄이기 위해 이진분할을 먼저 하고, 이진분할중 큰 서브클러스터를 또 이진분할하는 방법을 사용한다. 즉, 이진분할에서 시작하여 가장 큰 RE를 가지는 서브클러스터를 찾아 사용자가 원하는 t개의 서브클러스터가 생성될 때까지 반복 이진분할 한다.

2.2 개념 상승

데이터베이스의 일반화인 개념 상승은 각 튜플의 속성값을 관련 속성의 개념 트리에서 상위수준의 개념으로 대치시킴으로써 수행된다[7]. 일반화 시키고자 하는 수준은 응용별 개념 계층에 따라 다르다. 개념 계층의 상승은 데이터베이스가 일반화된 고수준의 개념을 가지며, 이때 중복되는 튜플은 합병하여 튜플 수를 줄인다.

예를 들어 표 1과 같이 조건속성이 내신, 수능, 그리고 결정속성이 졸업평점인 개념상승 관계가 있을 때, 중복수는 중복 튜플수를 나타낸다.

개념이 상승된 일반화 데이터베이스에서 결정규칙을 도출할 때 조건속성은 동일한데 결정속성이 상이한 모순된 결정규칙이 생성되는 현상 즉, 결정속성에 대한 조건속성의 충돌이 발생할 수 있다. 이를 해결하기 위한 방법은 첫째, 충돌이 발생한 튜플을 모두 제거하는 것인데, 이는 정보의 손실에 의하여 일부 규칙만 생성된다. 둘째, 확률이 적은 튜플을 제거하는 것인데, 이는 규칙이 한쪽으로 편향되어 신뢰성이 결여된 규칙이 유도된다. 본 논문에서는 이를 모두 수용하는 방법을 사용한다. 즉, 모순된 두개 이상의 규칙을 두개 이상의 결정속성값을 가지는 하나의 규칙으로 처리하여 각각의 결정속성값에 확률을 부여한다.

표 1. 개념상승 관계

튜플	내신	수능	졸업평점	중복수
1	수	수	상	35
2	우	수	상	40
3	우	우	상	25
4	우	수	중	50
5	수	미	중	70
6	우	미	중	60
7	수	수	중	2

q를 일반화 튜플, C_j를 목적 클래스라고 하면, q에 대한 확률 Prob는 q에 의한 목적 클래스를 구성한 원 튜플의 수와 q와 동치인 모든 클래스에 있는 튜플 총 수의 비율이다.

$$\text{prob} = \frac{count(q \subset C_j)}{\sum_{i=1}^{K} count(q \subset C_i)}$$

count : 중복 튜플의 수,
K : 클래스의 수, C_j : $\{C_1,...,C_k\}$

표 1에서 1번 튜플은 7번 튜플과 조건속성이 동일하므로 확률은 35/(35+2) = 95%이며, 2번은 4번과 같으므로 40/(40+50) = 44%, 3번은 25/25 = 100%이다.

개념 상승시 고려해야 할 또다른 문제는 빈도가 매우 적은 튜플의 처리인데, 이를 일반화 규칙으로 유도했을 때 규칙의 신뢰도가 매우 낮을 가능성이 있다. 따라서, 개념상승 관계에서 거의 나타나지 않는 튜플은 예외사항으로 간주하여 규칙의 일반화 이전에 사용자가 정한 잡음 필터 임계치보다 작을 때 제거한다.

q가 일반화 튜플일 때, q의 빈도율 Freq는 q의 중복 튜플 수와 총 튜플 수의 비율이다.

$$\text{Freq} = \frac{count(q)}{\sum_{i=1}^{n} count(q_i)}$$

잡음 필터 임계치는 일반화 관계에 있는 예외사항(매우 작은 빈도의 튜플)을 걸러내는 작은 값의 백분율이다. 표 1에서 튜플 7은 빈도율이 2/282 = 0.7%이므로, 잡음 필터 임계치가 1%라면 이 튜플은 삭제한다.

개념 상승은 생성한 개념 트리를 사용하여 데이터베이스를 개념 상승한 일반화 데이터베이스로 변환한다. 각 속성별로 개념 상승을 실행하며, 상승 수준은 사용자가 정한 상승수준 임계치까지 반복한다. 모든 속성에 대하여 개념 상승이 완료되면 중복 튜플을 조사하여 합병하고 중복 튜플 수를 누적하며, 전체적인 중복 튜플의 빈도를 계산하여 사용자가 선정한 잡음 필터 임계치보다 적은 튜플은 삭제한다.

3. 라프 셋과 속성 감축

3.1 라프 셋

라프셋은 식별불가능(indiscernible) 객체의 클래스로 구성된 동치관계를 기본으로 한다[12]. 본 논문에서는 일반 지식의 발견이 아닌 일반화 규칙의 발견을 다루므로 객체라는 용어 대신에 결정규칙이라는 용어로도 사용한다.

본 논문에서 결정규칙 시스템 S는 다음과 같이 정의한다.

S = {U, A, V}
U = {x_1, x_2, ...x_n}인 결정규칙의 유한집합
A = C∪D로서, C는 조건속성이고, D는 결정속성
V = $\bigcup_{p\in A}$Vp이며, Vp는 속성 p의 정의역

예를들어 U = {1, 2, 3, 4, 5, 6, 7, 8}, C = {a, b, c, d}, D = {e}, Va = Vd = Ve = {0, 1}, Vb = Vc = {0, 1, 2}일 때, 결정규칙 시스템을 표 2와 같은 결정규칙 테이블로 나타낼 수 있다.

P⊂A, x_i, x_j∈U일 때 U의 동치관계 R은 {(x_i, x_j) ∈U×U : for every p∈P, $p(x_i) = p(x_j)$}로 정의한다[2]. 즉, S에 있는 속성집합 P에서 iff $p(x_i) = p(x_j)$ for every p∈P일 때 x_i와 x_j는 동치이며, x_i의 동치 클래스는 $[x_i]_R$로 표현한다.

P⊆R이면 ∩P 또한 동치관계로서 IND(P)로 표현하며 P에 대한 식별불가능 관계라 한다[12].

S = {U, A, V}에서 U의 요소 x_i에 대해 동치관계 R의 동치클래스를 A의 기본집합이라 하며, X⊂U일 때, X의 R-긍정영역은 다음과 같이 정의한다[12].

$$POS_R(X) = \{x_i \in U \mid [x_i]_R \subset X\}$$

긍정영역은 지식 R을 사용하여 집합 X의 원소로 확실히 분류되는 객체의 집합이다.

P, Q⊂R에서, 결정속성 Q는 조건속성 P에 k(0≤k≤1)만큼 종속될 때, 종속도 k는 다음과 같이 정의한다[12].

표 2. 결정규칙 테이블

U	a	b	c	d	e
1	0	0	1	0	0
2	1	0	2	1	1
3	1	1	1	0	0
4	0	2	1	1	1
5	1	2	1	0	1
6	1	0	1	0	0
7	1	2	2	1	1
8	0	0	2	1	1

$$k(P,Q) = \frac{card\ POS_P(Q)}{card U}$$

k=1이면 Q는 P에 완전히 종속되며, 그렇지 않으면 부분적으로 종속이다. 따라서, k는 $P \Rightarrow_k Q$인 결정 테이블의 품질 척도라 할 수 있다. 표 2에서 P = {a, b, c, d}, Q = {e}라 하면 P에 대한 Q의 종속도는 1(k = 8/8)이고, P = {a, b}, Q = {e}이면 0.5(k = 4/8)이다.

속성간 종속도는 결정규칙 집합의 품질척도로 사용되지만 속성 감축의 기준으로도 이용된다.

3.2 속성 감축

결정규칙 시스템에서 속성간의 관계를 분석하여 불필요한 속성을 발견하면 이를 제거함으로써 간략화할 수 있다. 속성 감축은 불필요한 속성을 제거하고 전체 속성집합과 같은 품질 척도를 갖는 최소의 부분 속성집합으로 정의한다[9].

S = {U, A, V}에서 A=C∪D이고 B⊂C일 때, $POS_B(D) = POS_{B-\{p\}}(D)$라면 D에 대해 속성 p∈B는 B에서 불필요(dispensable) 속성이고, 그렇지 않으면 필요(indispensable) 속성이다. 모든 p∈B가 필요 속성이면 B는 독립이다.

특정 속성이 규칙 시스템에서 불필요하다면 원 시스템의 종속관계에 영향을 주지 않고 규칙 시스템에서 제거할 수 있다. D에 대해 C에 있는 필수 속성집합이 C의 core로, 속성 감축에서 제거할 수 없는 속성이다.

$$CORE(C, D) = \{a \in C \mid POS_C(D) \neq POS_{C-\{a\}}(D)\}$$

감축은 동치관계 R을 변화시키지 않으므로 결정규칙에 아무런 영향을 주지 않는다. 이는 감축된 집합으로 규칙 생성에 사용하더라도 X⊆U에 관련된 분류의 정확도가 변하지 않으므로, 원래의 속성집합보다 감축을 사용하는 것은 더 간결한 결정규칙을 생성할 수 있음을 의미한다.

표 2에서 C = {a, b, c, d}, D = {e}일 때

$POS_C(D) = POS_{C-\{a\}}(D)$
$POS_C(D) \neq POS_{C-\{b\}}(D)$
$POS_C(D) = POS_{C-\{c\}}(D)$
$POS_C(D) = POS_{C-\{d\}}(D)$

임으로 CORE(C, D) = {b}이다. 따라서 감축의 가능성은 {b}, {a, b}, {b, c}, {b, d}, {a, b, c}, {b, c, d}, {a, b, d}, {a, b, c, d}인데, 이를 모두 조사해 보면 {b, c}, {b, d}만이 D에 대해 독립이므로 감축 RED(C, D)

= {{b, c}, {b, d}}이다.

n개의 조건을 가진 규칙은 최고 2^n-1개의 부분집합에 대한 감축 가능여부를 조사해야 하는데, 이는 지수적 시간 복잡도를 가질 뿐 아니라, 많은 감축 중 어느 것이 가장 좋은 것인지 판단할 수가 없다. 실제 많은 응용에 있어서 한 개 또는 몇 개의 감축만 있어도 되는 경우가 많으며, 또한 가장 좋은 감축을 찾아낼 수 있다면 이것으로 최적의 결정규칙을 도출할 수 있다. 따라서, 본 논문에서는 모든 경우의 조사가 아닌 휴리스틱(heuristic)한 방법으로 좋은 감축을 찾아내는 방법을 사용한다. 이 방법은 Cercone[1]의 연구를 기반으로 한 것으로, 가장 좋은 감축을 찾아내기 위해서 속성의 중요도 순서로 부분집합을 구성하여 가장 먼저 감축으로 형성되는 속성집합을 최적 감축으로 판단한다.

속성의 중요도를 계산하는 방법에는 통계학에서 사용하는 χ^2 적합도 검증, 결정트리에 의한 기계학습에서 사용하는 정보획득량 측정[10], 라프셋 이론에서의 속성간 종속도 계산 방법 등이 있는데, 이중 정보획득량 측정방법이 규칙의 발견에 있어서 우수하므로[21], 본 논문에서는 이 방법을 사용한다.

감축은 조건속성 집합에서 결정속성에 필수적인 core 속성을 추출하고 core 속성을 포함하는 속성의 조합에 의하여 감축을 계산하는 집합론적 방법을 사용하나, 본 논문에서는 계산의 효율성을 위해 종속도를 조사하는 방법을 사용한다.

S = {U, A, V}, A = C ∪D에서 조건속성 C에 대한 결정속성 D의 종속도가 k(C, D)일 때 k(C, D) = k(C−{a}, D)이면 a∈C는 불필요하고 그렇지 않으면 필수 속성이므로, B⊆C, a∈B일 때 다음 조건을 만족하면 B는 C의 감축이다[1].

$$k(B, D) = k(C, D) \text{ and } k(B, D) \neq k(B-\{a\}, D)$$

감축을 계산하기 위한 필수 core 속성을 추출한다. core 속성을 결정하면 먼저 이를 감축집합으로 하고 core가 아닌 속성에 대해 중요도 순으로 감축집합에 추가하여 종속도의 증가를 측정한다. 원 데이터베이스의 종속도와 비교하여 동일한 종속도를 가지면 중지하고, 같지 않으면 계속 다음 속성을 추가하여 동일한 종속도를 가질 때까지 반복한다. 그리고 생성된 감축 속성집합에서 속성을 하나씩 제거하여 종속도에 변화가 있는지를 조사하여 종속도에 영향을 주지 않는 속성이 있으면 제거하여 최종 감축을 생성한다. 속성의 중요도 순으로 감축 집합을 생성하므로 첫 번째 생성되는 감축 집합이 최적 집합이다. 각 감축 집합에서 중복 튜플이 발생하면 합병하고 중복 튜플수를 누적시킨다.

속성 및 속성값을 효율적으로 감축하는데 식별가능(discernible) 행렬과 식별가능 함수[15,19]를 사용할 수 있다. 따라서 본 논문에서는 계산의 효율을 위해 식별가능 행렬과 함수를 이용하여 core 속성을 추출하고 또 속성값을 감축하여 최소 결정 테이블을 유도한다.

4. 실험 및 평가

4.1 실험

본 논문에서는 표 3과 같이 98년 계명대학교 공과대학을 졸업한 학생들에 대하여 입학성적과 졸업평점을 중심으로 한 일반화 규칙을 유도해 보고자 한다.

개념 트리의 생성에 필요한 속성은 출신지역, 내신점수, 수능점수, 졸업평점으로, 출신지역을 제외한 속성들은 수치이므로 자동으로 개념트리 생성이 가능하다. 출신지역은 기존의 지식에 의하여 수동적으로 대도시, 중도시, 소도시로 구분하여 생성하고, 내신점수, 수능점수, 졸업평점은 클러스터링 방법에 의해 상, 중, 하로 자동생성한다. 자동 생성된 개념 트리는 표 4와 같다.

93년 이전 입학학생과 94년 이후 입학학생의 내신점수와 수능점수는 체계가 달라 분리하여 개념트리를 생성했다.

표 3. 졸업생 데이터베이스(일부)

번호	입학년도	출신지역	재수구분	내신점수	수능점수	면접등급	졸업평점
45	1993	대구	재수	141.7	280.0	A	3.6713
46	1993	대구		139.7	275.0	C	4.1389
47	1993	대구	재수	137.7	261.0	A	3.2862
48	1993	경산		143.7	255.0	C	2.4610
49	1994	대구	재수	145.8	395.0	A	4.0347
50	1994	대구		143.2	360.0	A	3.1923
51	1994	대구		134.4	370.0	A	3.7431

표 4. 생성된 개념 트리

<출신지역>
포항, 경주, 대구, 부산 → 대도시
김천, 구미, 경산, 상주 → 중도시
칠곡, 현풍, 성주, 문경, 청도, 거창, 제천 → 소도시
<내신점수>
93이전:142~146 → 상, 140~140 → 중, 134~138 → 하
94이후:124~146 → 상, 100~120 → 중, 072~092 → 하
<수능점수>
93이전:255~280 → 상, 233~251 → 중, 203~231 → 하
94이후:385~400 → 상, 365~380 → 중, 331~360 → 하
<졸업평점>
3.4~4.1 → 상, 2.9~3.3 → 중, 2.2~2.8 → 하

사례중 적어도 2개 이상의 사례가 일반화 규칙으로 지지를 받을 수 있을 것으로 판단되어 잡음 필터 임계치는 2%로 설정한다. 몇 개의 튜플에서 조건속성이 동일하나 졸업평점이 상이한 모순이 발생하는데, 이는 생성되는 규칙에 확률을 표시한다. 개념 상승된 일반화 데이터베이스는 표 5와 같다.

졸업평점을 결정속성으로 하고, 졸업평점과 관련된 규칙을 유도하기 위해 출신지역, 재수구분, 내신점수, 수능점수, 면접등급을 조건속성으로 하여 식별가능 행렬로 표시하면 표 6과 같다.

식별가능 행렬에서 단일 속성은 내신점수(C)와 수능점수(D)인데, 이것이 감축의 core이다. 그리고 core 속성을 제외한 속성인 출신지역, 재수구분, 면접등급의 중요도를 정보획득량 측정방법에 의하여 계산하면 다음과 같다.

면접등급 : 0.419
출신지역 : 0.303
재수구분 : 0.088

재수구분은 졸업평점에 거의 영향을 미치지 않음을 알 수 있다.

core 속성과 속성의 중요도를 사용하여 일반화 데이터베이스를 감축하면 2개의 감축 {내신점수, 수능점수, 면접등급}, {출신지역, 내신점수, 수능점수}이 생성된다.

표 5. 개념상승된 일반화 데이터베이스

	출신 지역 (A)	재수 구분 (B)	내신 점수 (C)	수능 점수 (D)	면접 등급 (E)	졸업 평점 (F)	중복수
1	대도시	재수	상	상	A	상	5
2	대도시	일반	상	중	A	상	9
2'	대도시	일반	상	중	A	중	4
3	소도시	일반	상	하	B	상	5
4	대도시	일반	상	하	A	중	5
5	대도시	일반	중	중	B	상	5
6	대도시	일반	중	하	C	상	3
7	중도시	일반	중	하	B	하	5
8	대도시	재수	하	하	A	중	3
9	대도시	일반	하	하	C	중	5
9'	대도시	일반	하	하	C	하	2

표 6. 식별가능 행렬

	1	2	3	4	5	6	7	8
1								
2	BD							
3		ADE						
4	BD	D	AE					
5		CE		CDE				
6		CDE		CE				
7	ABCDE	ACDE	AC	ACE	AD	AE		
8	C	BCD	ABCDE		BCDE	BCDE	ABCDE	
9	BCDE	CDE	ACE	CE	CDE	C	ACE	BDE

표 7. 최소 결정규칙 테이블

규칙	출신 지역	내신 점수	수능 점수	졸업 평점	중복수
1		상	상	상	5
2		상	중	상	9
2'		상	중	중	4
3	소도시			상	5
4	대도시	상	하	중	5
5	대도시	중		상	8
6	중도시			하	5
7		하	상	중	3
8		하	하	중	5
8'		하	하	하	2

본 실험에서는 감축 {출신지역, 내신점수, 수능점수}로부터 일반화 규칙을 유도하는데, 식별가능 행렬과 함수를 사용하여 불필요 속성값을 제거한 최소 결정규칙 테이블은 표 7과 같다.

여기서 최소 규칙을 언어변수로 표현하고, 모순된 규칙에 대해서는 확률까지 나타낸 최종 일반화 규칙은 다음과 같다.

규칙 1 : IF (내신점수 = 상 and 수능점수 = 상) or (출신지역 = 소도시) or (출신지역 = 대도시 and 수능점수 = 중) THEN 졸업평점 = 상
규칙 2 : IF (내신점수 = 상 and 수능점수 = 중) THEN 졸업평점 = 상(69%), 중(31%)
규칙 3 : IF (출신지역 = 대도시 and 내신점수 = 상 and 수능점수 = 하) or (내신점수 = 하 and 수능점수 = 상) THEN 졸업평점 = 중
규칙 4 : IF 내신점수 = 하 and 수능점수 = 하 THEN 졸업평점 = 중(71%), 하(29%)
규칙 5 : IF (출신지역 = 중도시) THEN 졸업평점 = 하

상기 유도된 결정규칙은 실험 데이터가 한개 특정 대학의 자료를 분석한 것이므로 생성된 규칙이 보편적으로 적용될 수 있는 일반화 규칙으로 보기는 어렵다. 만약 실험 데이터가 신뢰성이 높고 그 양이 많다면 일반적이고도 신뢰성 있는 규칙이 유도될 것이다.

4.2 평가

본 논문에서 제안한 방법을 속성중심 귀납법[7]과 라프셋에 의한 지식감축 방법[12]으로 유도한 결과와 비교하여 평가한다.

속성중심 귀납법은 저수준 데이터베이스를 개념 상

표 8. 결정규칙 유도방법의 비교

특성 \ 구분	속성중심 귀납	라프셋의 지식 감축	본 논문
규칙의 간략성	낮음	높음	높음
규칙의 추상성	높음	낮음	높음
규칙의 중복성	있음	없음	없음
규칙의 최소화	불가능	가능	가능
감축의 적합성 판단	N/A	불가능	가능
계산시간 복잡도	N/A	$O(2^n)$	$O(n^2)$

승만시켜 일반화된 지식표현 시스템으로 표현하고, 그것을 바로 언어변수를 사용한 규칙으로 변환하므로, 최종 규칙으로 보기에 너무 복잡하고 중복된 지식을 포함하여 간결성이 없다[20].

저수준의 데이터베이스에 라프셋의 지식감축 방법을 직접 적용하는 경우는 속성과 속성값의 감축에 소요되는 시간이 지수적 복잡도를 가진다. 그리고 생성된 감축중에서 어느 감축이 가장 좋은지 판단할 수가 없으며, 또한 실제 결정규칙을 도출했다 하더라도 저수준의 표현이기 때문에 지식으로서의 추상성이 없다[20].

본 논문에서 제안한 방법은 데이터베이스를 일반화시켜 추상성을 높일 뿐만 아니라, 속성의 중요도를 고려하여 감축을 생성하므로 감축 속도가 $O(n^2)$의 시간 복잡도를 가지며, 또한 감축의 적합성을 판단할 수 있다. 그리고 속성값간의 관계를 조사하여 불필요한 속성값을 제거함으로써 최소화 결정규칙을 도출한다. 이 결정규칙은 실험 결과에서 본 바와 같이 매우 간결하며 지식이 고수준의 추상으로 표현되어 그 의미의 이해와 해석이 용이하다.

세가지 방법의 장단점을 요약하여 비교하면 표 8과 같다.

5. 결 론

본 논문에서는 특정 영역에 대한 지식을 일반화하고, 불필요한 사항을 제거하여 최소 결정규칙을 유도하였다. 이를 위해 클러스터링에 의한 개념트리 생성의 자동화, 개념 상승에 의한 데이터베이스의 일반화, 정보획득량 측정에 의한 속성의 중요도 계산, 중요도를 이용한 속성 감축에 의한 최적 감축, 식별가능 행렬을 이용한 효율적인 속성값 감축을 연구했다.

본 연구에서 제안한 방법은 첫째, 대규모 데이터베이스에 내재된 중요한 규칙을 발견하므로, 각종 투자계획, 가격결정 등과 같은 의사결정 업무에 적용될 수 있다. 둘째, 데이터로부터 최적의 규칙을 유도하므로, 각종 고장진단, 의료진단 등의 전문가 시스템을 위한 지식베이스의 구축에 유효하게 사용될 수 있다. 셋째, 시장분석, 실험자료 분석 등 각종 데이터 분석에 이용될 수 있다. 그리고 데이터베이스의 정보검색에 있어서, 고수준 개념의 질의처리에 유용하게 이용된다. 즉, 언어변수를 사용한 고수준의 질의를 개념 계층을 하향식으로 적용하여 적합한 튜플 집합을 검색할 수 있다.

지식 감축에 사용된 방법은 작은 시간 복잡도를 가지며, 감축된 지식이 최소 규칙으로 간략화되므로 의사결정 뿐 아니라, 데이터 분류, 데이터 요약, 분류규칙의 학습 등 여러 분야에 적용될 수 있다.

본 논문에서는 모순된 규칙에 대해서 확률이라는 정량적 값을 표시하거나 잡음 필터를 사용하여 신뢰성이 적은 규칙을 제거하는 방법을 사용하였으나, 이는 완벽한 해결책이라 볼 수 없다. 따라서 향후 과제로는 불완전한 데이터를 더 효과적으로 처리하는 방법을 연구해야 하며, 또한 유도된 규칙을 훈련 데이터로 계속 정련하여 신뢰성을 향상시키는 방법을 연구해야 한다.

참고문헌

[1] N. Cercone, H. Hamilton, X. Hu and N. Shan, "Data Mining Using Attribute-Oriented Generalization and Information Reduction," *Rough Sets and Data Mining,* T. Lin and N. Cercone (eds), Kluwer, pp. 199-227, 1997.

[2] D. Cheung, A. Fu and J. Han, "Knowledge Discovery in Databases: A Rule-Based Attribute-Oriented Approach," 1988.

[3] W. Chu, H. Yang, K. Chiang, M. Minock, G. Chow, and C. Larson, "CoBase: A Scalable and Extensible Cooperative Information System," *Intelligent Integration of Information,* G. Wiederhold (eds), JIIS, Vol. 6, No. 2/3, pp. 223-259, 1996.

[4] D. Fisher, "Knowledge Acquisition via Incremental Conceptual Clustering," *Machine Learning*, vol. 2, no. 2, pp. 139-172, 1987.

[5] W. Frawley, G. Piatetsky-Shapiro and C. Matheus, "Knowledge Discovery in Databases: An Overview," G, Piatetsky-Shapiro and W. Frawley (eds), *Knowledge Discovery in Databases,* AAAI/MIT Press, pp. 1-27, 1991.

[6] D. Fudger and H. Hamilton, "A Heuristic for Evaluating Databases for Knowledge Discovery with DBLEARN," *Proc. RSKD'93,* Banff, Alberta, Canada, pp.32-43, 12-15 Oct., 1993.

[7] J. Han, Y. Cai, and N. Cercone, "Knowledge Discovery in Databases: An Attribute-Oriented Approach," *Proceeding of the 18th Conference on Very Large Data Bases,* Vancouver, Canada, pp. 340-355, 1992.

[8] X. Hu, N. Cercone, and J. Han, "An Attribute-Oriented Rough Set Approach for Knowledge Discovery in Databases," *Proc. RSKD'93,* Banff, Alberta, Canada, pp. 90-99, 12-15 Oct., 1993.

[9] X. Hu, N. Cercone, and W. Ziarko, "Generation of

Multiple Knowledge from Databases Based on Rough Set Theory," *Rough Sets and Data Mining*, T. Lin and N. Cercone (eds), Kluwer, pp. 109-121, 1997.

[10] M. Kamber, L. Winstone, W. Gong, S. Cheng and J. Han, "Generalization and Decision Tree Induction: Efficient Classification in Data Mining," http://www.kdnuggets.com/, 1999.

[11] M. Kryszkiewicz and Henryk Rybinski, "Finding Reducts in Composed Information Systems," *Proc. RSKD'93,* Banff, Alberta, Canada, pp. 261-273, 12-15 Oct., 1993.

[12] *Z. Pawlak, Rough Sets – Theoretical Aspects of Reasoning about Data,* Kluwer, 1991.

[13] Z. Pawlak, *"Rough Sets," International Journal of Computer and Information Science,* 11, pp. 341-356, 1982.

[14] J. Quinlan, "Induction of Decision Trees," *Machine Learning*, Vol. 1, No. 1, pp. 81-106, 1986.

[15] A. Skowron and C. Rauszer, "The Discernibiity Matrices and Functions in Information Systems," Slowinski (eds), *Intelligent Decision Support Handbook of Advances and Applications of the Rough Set Theory*, Kluwer, pp. 311-362, 1991.

[16] S. Wong and W. Ziarko, "On Optimal Rules in Decision Tables," *Bulletin of The Polish Academy of Sciences Mathematics,* Vol. 33, No. 11-12, 1985.

[17] Y. Xiang, S. Wong, and N. Cercone, "Quantifying Uncertainty of Knowledge Discovered from Databases," *Proc. RSKD'93,* Banff, Alberta, Canada, pp. 63-73, 12-15 Oct., 1993.

[18] W. Ziarko, "Rough Sets and Knowledge Discovery: An Overview," *Rough Sets, Fuzzy Sets and Knowledge Discovery, Proc. RSKD'93,* Banff, Alberta, Canada, pp. 11-13, 12-15 October, 1993.

[19] 이성주, 정환묵, 최완규, 러프집합과 응용, 조선대학교 출판국, 1998.

[20] 정홍, 정환묵, "지식 발견을 위한 라프셋 중심의 통합 방법 연구," 퍼지 및 지능 시스템학회 논문지, 제8권, 제6호, pp. 27-36, 1998.

[21] 정홍, 최경욱, 정환묵, "Generation of Approximation Rules Using Information Gain," *FUZZ-IEEE '99, The 8th Int'l Conf. on Fuzzy Systems,* Seoul, Korea, 22-25 Aug., pp. 1241-1245, 1999.

[22] 지원철, 김민용, "데이터마이닝과 의사결정지원 시스템," 정보과학회지, 제16권, 제9호, pp. 24-36, 1998.

정 홍 (Hong Chung)

1972년 : 한양대학교 원자력공학과 (공학사)
1976년 : 고려대학교 경영대학원 (경영학 석사)
1996년 : 대구효성가톨릭대학교 전산통계학과 (이학석사)
1999년 : 대구효성카톨릭대학교 전산통계학과(이학박사)

1972~1981년 : 한국과학기술연구원 선임연구원
1981년~현재 : 계명대학교 컴퓨터전자공학부 부교수
주관심분야 : 지능정보 시스템, 소프트웨어 공학

정 환 묵(Hwan-Mook Chung)

제 9권 제 3호 참조