지지도와 신뢰도의 가중치에 기반한 분류알고리즘에 관한 연구

김 근 형*

Study on Classification Algorithm based on Weight of Support and Confidence Degree

Keun-Hyung Kim*

이 논문은 2007년도 제주대학교 학술연구지원사업에 의하여 연구되었음
(This work was supported by the research grant of the Cheju National University in 2007)

요 약

데이터마이닝 분야에서 기존의 분류알고리즘들은 보다 적은 컴퓨팅 자원을 이용하여 보다 빨리 분류모형을 생성하고자 하는 효율성 중심의 연구가 주를 이루었다. 본 논문에서는 분류알고리즘의 효율성을 추구할 뿐 아니라 온톨로지 자동생성이나 비즈니스 환경 등 각 응용분야에 적합한 유효한 분류규칙을 보다 많이 생성할 수 있는 효과성도 동시에 추구하였다. 이를 위하여 지지도와 신뢰도의 가중치가 적용된 가중치적용함수를 제안하였고 이 함수의 성질들을 이론적으로 규명하였다. 가중치적용함수를 사용하면서 새로운 분리기준 설정방법을 제안하였고 또한 새로운 분류알고리즘을 제안하였다. 제안한 알고리즘의 성능평가 결과 기존의 우수한 알고리즘보다 보다 많은 유효한 분류규칙들을 보다 신속하게 생성함을 알 수 있었다.

ABSTRACT

Most of any existing classification algorithm in data mining area have focused on goals improving efficiency, which is to generate decision tree more rapidly by utilizing just less computing resources. In this paper, we focused on the efficiency as well as effectiveness that is able to generate more meaningful classification rules in application area, which might consist of the ontology automatic generation, business environment and so on. For this, we proposed not only novel function with the weight of support and confidence degree but also analyzed the characteristics of the weighted function in theoretical viewpoint. Furthermore, we proposed novel classification algorithm based on the weighted function and the characteristics. In the result of evaluating the proposed algorithm, we could perceive that the novel algorithm generates more classification rules with significance more rapidly.

키워드

분류알고리즘(Classification Algorithm), 효율성(Efficiency), 효과성(Effectiveness), 가중치(Weight), 지지도(Degree of Support), 신뢰도(Degree of Confidence), 온톨로지(Ontology)

* 제주대학교 경영정보학과 부교수

접수일자 2008. 11. 06

Ⅰ. 서 론

데이터마이닝은 트랜잭션시스템에 의하여 축적된 대규모의 데이터들을 분석함으로써 기업의 이윤추구에 도움이 될 수 있는 정보와 지식을 획득할 수 있는 기술이다. 데이터마이닝 기술은 다양한 분야에서 응용되고 있다. 복잡한 온톨로지를 생성하기 위한 기반 기술이 되기도 하고 비즈니스환경의 마케팅 프로세스에서도 중요하게 이용된다[1,2].

데이터마이닝 기법 중에서 자주 활용이 되고 또한 가장 활발히 연구되어져 왔던 분야가 연관규칙탐사(association rule mining)와 의사결정나무(decision tree) 모형에 의한 분류(classification) 분야이다.

연관규칙탐사기법에 관한 지금까지의 연구들은 효율성(예를 들면, 정보처리의 속도)과 효과성(예를 들면, 보다 유의미한 정보 또는 지식 추출) 측면을 균형있게 고려하면서 연구되어져 왔다. 연관규칙탐사기법의 초창기 연구에서는 주로 효율성을 목적으로 한 신속한 정보처리를 중요시 하는 알고리즘들이 제안되었으나[3,4] 이 후에는 효과성 측면의 관점에서 비즈니스 요구사항에 부응할 수 있는 유의미한 규칙을 탐사할 수 있는 알고리즘들이 제안되었다[5,6,7]. 특히, 데이마이닝 기술을 온톨로지의 자동생성기술에 응용하기 위해서는 내용중심의 효과성 지향 기술에 대한 개발이 더욱 필요하다.

반면, 의사결정나무모형(Decision Tree)을 생성하기 위한 분류기법(Classification Technique)에 관한 지금까지의 연구들은 효율성 중심의 연구들로 편향되었다고 할 수 있다. 메모리제약을 극복하면서 대용량의 데이터를 처리하기 위한 병렬알고리즘에 대한 연구[8]라든지, 모형생성단계(Building Phase)와 가지치기단계(Pruning Phase)의 통합을 통하여 의사결정나무 모형을 보다 신속하게 생성하는 기법[9]이라든지, 신속하게 의사결정나무 모형을 생성하기 위하여 데이터베이스에 대한 접근횟수를 최소화하기 위한 기법[10] 등 분류기법에 대한 대부분의 연구들이 비즈니스 요구사항을 고려하지 않은 시스템 효율 중심의 연구였다. 이러한 효율성 중심의 연구들은 컴퓨터 하드웨어 기술의 발전으로 인하여 그 가치가 점점 퇴색될 것이다.

그러나, 데이터마이닝의 분류기법을 비즈니스 분야에서 응용하고자 할 때 효과성을 더 중시해야 하는 측면이 있다. 예를 들면, 마케팅 분야에서 고객데이터를 기반으로 고객유형을 분류하여 각 그룹별 특성을 파악하고자 할 때 각 그룹의 크기(세그먼트 크기)가 일정정도 이상의 규모가 되어야 한다. 마케팅의 시장세분화 전략이 성공하기 위하여 세분된 세그먼트의 크기가 수익성을 끌어낼 수 있을 정도의 충분한 크기여야 한다는 것은 마케팅이론의 정설이다[11]. 예를 들어, 관광지를 방문한 관광객 중에서 재방문할 가능성이 있는 관광객의 특성을 분석하여 이들만을 대상으로 한 타겟 마케팅(target marketing)을 수행 하고자 할 때, 관광객 특성에 의한 재방문 가능성이 있는 관광객 그룹의 크기는 일정정도 이상이 되어야 마케팅의 의미가 있다는 것이다.

분류기법에 대한 지금까지의 연구결과를 보면 세그먼트의 크기를 고려하지 않고 의사결정나무모형 즉, 분류모형을 생성하고 있으며 이로 인하여 과잉맞춤(over-fitting)문제가 발생될 뿐만 아니라 도출된 분류모형이 훈련데이터(training data set)에 의존하게(dependent) 되는 결과를 초래한다. 기존 연구들은 과잉맞춤문제를 해결하기 위하여 분류모형(의사결정 나무구조)내 서브트리(sub tree)의 암호화용량을 비교하는 물리적 방법을 취하고 있기 때문에[9] 가지치기를 거친 최종적인 분류모형이라 할 지 라도 여전히 훈련데이터에 의존적인 분류모형이 된다. 이는 결국 현실세계의 실제 데이터를 분류하는 정확도를 떨어뜨리는 이유가 된다. 분류모형을 생성할 때 세그먼트의 크기를 고려한다면 과잉맞춤문제는 자연적으로 해결될 뿐만 아니라 최종적인 분류모형도 훈련데이터에 덜 의존적이게 되어서 현실세계의 실제 데이터를 분류하는 정확도는 높아질 것이다.

분류기법에 대한 기존 연구들의 또 다른 문제는 그 처리방법의 구조적 한계로 인하여 비즈니스적 가치가 있는 유의미한 분류규칙을 간과한다는 점이다. 기존의 분류기법은 물리적 기준인 엔트로피(entropy) 값을 최대화하는 속성과 속성값을 선택하여 의사결정나무의 분리속성(splitting attribute)으로 선택한다. 이는 현실세계의 실제 데이터를 더 정확히 분류할 가능성이 있는 또 다른

속성 특성을 간과해버리는 결과가 될 수 있다. 예를 들면, 100개의 레코드로 이루어진 훈련데이터에서 지역속성 값이 '강남'인 레코드들의 수는 50개이고 이중에서 48개 레코드의 응답속성 값이 '예'인 경우와 직업속성 값이 '자영업'인 레코드들의 수가 2개이고 이 중에서 2개 레코드의 응답속성 값이 '예'인 경우, 엔트로피 값에 의한 기존 분류기법은 직업 속성과 속성값 '자영업'을 분리기준으로 선택할 가능성이 높다. 그러나 통계학적으로는 지역속성이 현실세계의 실제 데이터를 더 잘 분류할 가능성이 있다고 할 수 있다.

본 논문에서는 세그먼트크기를 고려하여 분리속성을 결정하는 새로운 분류알고리즘을 제안한다. 새로운 분류알고리즘은 비즈니스적 가치가 있는 보다 많은 유의미한 분류규칙들을 보다 신속하게 생성함으로써 효율성과 효과성을 동시에 만족시킬 수 있다.

2장에서는 분류알고리즘과 관련한 기존 연구들을 고찰하고 분석하며, 3장에서 새로운 분류알고리즘을 제안한다. 4장에서는 제안한 알고리즘을 기존의 우수한 알고리즘과 비교하여 그 성능을 평가 분석하며 5장에서 결론을 맺는다.

Ⅱ. 선행연구

1. 분류알고리즘과 용어정의

1.1 분류알고리즘 개요

분류알고리즘의 입력 데이터는 부류속성(class label)을 포함하는 레코드들로 구성된 훈련데이터(training data)이다. 레코드는 속성값들로 이루어지는데 속성값은 범주형(categorical)과 수치형(numerical)으로 나눌 수 있다. 분류알고리즘은 훈련데이터를 입력으로 하여 속성과 속성값들의 관점에서 부류속성을 설명하는 분류모형 즉 의사결정나무를 생성한다.

그림 1에서 (a)와 (b)는 대출승인응용을 위한 훈련데이터이고 (c)는 분류알고리즘이 이 훈련데이터를 입력으로 받아 생성한 분류모형이다.

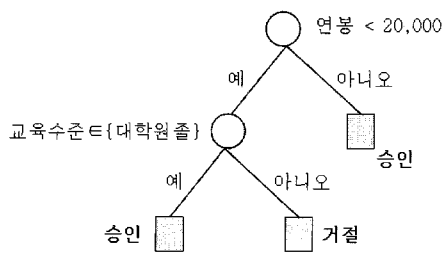
그림 1의 훈련데이터에서 각 레코드는 하나의 대출요청과 대응되고 세 개의 속성을 갖는다. 세 속성 중 승인여부라는 속성은 두 값 중 하나의 값만을 갖는 부류속성을 갖는다. 승인여부라는 부류속성은 대출신청이 허락되면 '승인' 값을 갖고 불허되면 '거절'값을 갖는다. 연봉이라는 속성은 대출신청자의 수입과 관련된 속성으로써 수치형 데이터를 그 값으로 가지며 교육수준 속성은 {고졸, 대졸, 대학원졸}이라는 범주형 값들 중 하나의 값을 갖는다.

연봉	교육수준	승인여부
10,000	고졸	거절
40,000	대졸	승인
15,000	대졸	거절
75,000	대학원졸	승인
18,000	대학원졸	승인
22,000	대졸	승인

(a)

연봉	교육수준	승인여부
10,000	고졸	거절
40,000	대졸	승인
15,000	대졸	거절
75,000	대학원졸	승인
18,000	대학원졸	승인
22,000	대졸	거절

(b)



(c)

그림 1. 훈련데이터와 분류모형
(a) 훈련데이터A (b)훈련데이터B (c)분류모형
Fig. 1 Training Data and Classification Model
(a)training data A (b) training data B
(c) Classification Model

그림 1의 (c)는 이러한 훈련데이터에 의하여 생성된 분류모형을 나타내고 있다. 분류모형의 목적은 대출신청이 허락되거나 거절되는 조건을 대출신청자의 연봉과 교육수준의 관점에서 설명하는 것이다. 분류모형에서 원모양의 노드를 내부노드(internal node)라 하고 사각형모양의 노드를 잎노드(leaf node)라고 한다. 내부노드는 레코드의 특정 속성값에 따라 적절한 다른 노드로 분기할 수 있게 하는 역할을 한다. 잎노드는 레코드의 부류값을 나타낸다. 새로운 레코드가 분류모형으로 입력되면 루트노드(root node, 내부노드임)부터 시작하여 레코드의 특정 속성값들을 체크하면서 적절한 다른 내부노드로 분기되며 그 내부노드에서 다시 분기되는 과정을 반복적으로 거치면서 결국은 잎노드에 도달하게 되고 그 입력 레코드의 부류값을 예측할 수 있게 된다. 예를 들면, 그림1 (c)의 분류모형에서 '연봉 ≥ 20,000' 이거나 '교육수준 ∈{대학원}' 인 대출신청자에 대응하는 레코드는 '승인' 이라는 잎노드에 도달하게 된다.

분류모형은 분류규칙들의 모임이라고도 할 수 있다. 그림1 (c)의 분류모형은 3개의 분류규칙을 포함한다. 즉, 「'연봉< 20,000'이고 '교육수준'∈{대학원졸}'이면 대출을 승인한다」라는 규칙과 「'연봉< 20,000'이고 '교육수준'∉{대학원졸}'이면 대출을 거절한다」, 그리고 「'연봉≥ 20,000'이면 대출을 승인한다」라는 규칙을 포함한다.

특히 본 논문에서, 신뢰도(degree of confidence)는 분류모형의 내부노드 또는 잎노드에 대응하는 레코드들이 특정 부류에 속할 확률을 의미하며, 지지도(degree of support)는 훈련데이터의 전체 레코드 수에 대하여 내부노드 또는 잎노드에 대응하는 레코드 수의 상대적인 크기를 의미하는 것으로 정의한다. 예를 들어, 그림 1 (c)의 분류모형에서 훈련데이터A 내의 '연봉 ≥20,000'인 모든 레코드는 100%의 신뢰도로 그 부류속성값이 '승인' 값을 갖지만, 훈련데이터 B에서 대해서는 66%의 신뢰도를 갖는다. 반면, '연봉 ≥20,000'인 레코드에 대응하는 잎노드의 지지도는 훈련데이터A와 훈련데이터B 모두에서 50%가 된다.

비슷한 개념으로, 최소신뢰도(degree of minimum confidence)는 분류모형의 모든 잎노드들이 만족해야 하는 신뢰도를 의미하며, 최소지지도(degree of minimum support)는 분류모형의 모든 잎노드들이 만족해야 하는 지지도를 의미하는 것으로 정의하기로 한다. 최소지지도나 최소신뢰도는 사용자에 의하여 설정될 수 있다.

분류모형을 생성하기 위한 대부분의 분류알고리즘들은 생성단계(building phase)와 가지치기단계(pruning phase)의 2단계로 분류모형을 생성한다. 생성단계에서는 훈련데이터의 모든 잎노드가 동일부류(pure) 또는 일정 신뢰도 이상일 때까지 서브그룹으로 분할하는 과정을 순환적으로 반복하면서 분류모형을 생성한다. 처음에는 분류모형이 전체 훈련데이터에 대응하는 루트노드만을 포함하였다가 그 훈련데이터가 새로운 서브그룹들로 분할되면 각 서브그룹에 대응하는 새로운 노드가 분류모형에 추가되는 방식으로 분류모형이 만들어지는 것이다. 이때 서브그룹들로 분할하기 위한 분리기준이 필요한데, 일반적으로 특정속성과 속성값이 선택된다. 예를 들어, 그림1 (c)의 분류모형에서 루트노드의 분리기준은 분리속성이 '연봉'이고 속성값은 '20,000'이 된다. 일반적으로 서브그룹 P가 있을 때 분리기준 T에 의하여 새로운 서브그룹들 P1, P2,...,Pm이 생성되게 된다. 이때 P에 대응하는 노드는 분리기준이 T가 되며 서브그룹들 P1, P2,...,Pm도 순환적으로 새로운 분리기준에 의하여 새로운 자식노드들로 분할되게 된다.

가지치기 단계는 훈련데이터에 너무 의존적인 분류모형을 좀 더 일반화시키는 과정이다. 훈련데이터 내 극소수의 변칙 데이터에 대응되어 생성된 잎노드들을 과잉맞춤(over-fitting)이라고 하는데, 과잉맞춤에 의하여 만들어진 잎노드들은 현실세계의 실제 데이터와는 불일치할 가능성이 크기 때문에 분류모형에서 제거될 필요가 있다.

1.2 생성단계

일반적으로 생성단계는 모든 잎노드가 동일부류에 속할 때까지 순환 반복적으로 훈련데이터를 분할한다. 그림2는 분류알고리즘의 생성단계 프로시저를 나타내고 있다.

```
procedure buildTree(S):
1. Initialize root node using data set S
2. Initialize queue Q to contain root node
3. while Q is not empty do {
  dequeue the first node N in Q
  if N is not pure {
    for each attribute A
      Evaluate splits on attribute A
    Use best split to split node N
          into N1 and N2
    Append N1 and N2 to Q
  }}
}
```

그림2. 분류알고리즘의 생성단계
Fig.2 Generation Phase of Classification Algorithm

그림 2의 생성단계 프로시저에서, 훈련데이터 S나 서브그룹 N을 분할하기 위한 분리기준(splitting condition)의 형태는 속성 A가 수치형일 경우 'A < v'(v는 A의 도메인에 속함)의 형식이고, 속성 A가 범주형일 경우 'A ∈V'(V는 A의 도메인에 속하는 값들의 집합) 형식이다. 이때, 분리기준으로써 속성A와 그 값 v나 V를 결정하는 기준은 분할된 서브그룹의 엔트로피(entropy)값을 최소화할 수 있어야 한다. 레코드집합 S에 대하여 정보전달 효율척도인 엔트로피 $E(S) = -\sum_j p_j \log p_j$ (단, pj는 S를 구성하는 레코드들에 대해서 부류j에 속하는 레코드들의 상대적인 수)로 정의할 수 있다[9]. 따라서 S에 속한 레코드들이 동일부류에 속할 확률이 클수록 엔트로피 값은 작아진다. n개의 레코드들로 구성된 집합 S를, n1개의 레코드들로 이루어지는 집합 S_1과 n_2개의 레코드들로 이루어지는 집합 S_2로 분할할 경우, 엔트로피 $E(S_1, S_2) = \frac{n_1}{n}E(S_1) + \frac{n_2}{n}E(S_2)$를 최소화시키는 속성과 속성값이 분리기준으로 선택된다. 이는 결국 S를 S_1과 S_2로 분리할 때 S_1의 신뢰도와 S_2의 신뢰도의 합이 최대가 되게 하는 분리기준을 선택하는 것이라고 할 수 있다.

1.3 가지치기 단계

가지치기 단계에서는 분류모형내의 과잉맞춤을 제거하는 과정인데 일반적으로 MDL(Minimum Description Length)원리[12]를 이용한다. MDL의 기본 철학은 최소의 비트들로 암호화시킬 수 있는 분류모형이 가장 좋은 분류모형이라는 것이다. 따라서 가지치기 단계에서는 최소의 비트들로 암호화될 수 있는 서브트리들을 찾는 과정이라고 할 수 있다.

k개의 부류(class)들 중의 하나에 속하는 n개의 레코드들로 구성된 레코드집합 S가 있고, ni는 부류 i에 속하는 레코드들의 수라고 할 때 S의 레코드들의 부류들을 암호화시키는 비용 C(S)는 다음과 같이 계산할 수 있다[9].

$$C(S) = \sum_i n_i \log \frac{n}{n_i} + \frac{k-1}{2}\log\frac{n}{2} + \log\frac{\pi^{k/2}}{\Gamma(k/2)}$$

분류모형의 트리구조를 암호화시키기 위해서는 포함하고 있는 노드가 내부노드(비트 1)인지 잎노드(비트 0)인지를 구분하기 위하여 노드 당 하나의 비트를 필요로 한다. 따라서, 그림1의 (c)와 같은 분류모형의 경우 비트스트링 '11000' 형태로 암호화시킬 수 있다.

분류모형을 암호화시키기 위하여 분리기준 또한 암호화시킬 수 있어야 한다. 분리기준을 암호화시키기 위해서 분리속성과 속성값에 대한 암호화가 필요하다. 훈련데이터의 속성 갯수가 a개라면 분리속성을 암호화시키기 위하여 필요한 비트수는 log a 만큼 필요하다. v개의 값들로 이루어진 속성값을 암호화시키고자 할 때 수치형 데이터의 경우 $\log(v-1)$의 비트수가 필요하고 범주형 데이터의 경우 $\log(2^v-2)$의 비트수가 필요하다[9].

그림 3은 가지치기 단계의 처리과정을 순환적으로 나타내고 있다.

```
procedure Pruning(Node N):
/* S is the set of data records for N */
if N is a leaf return(C(S)+1)
/* N1 and N2 are N's children */
minCost1 := computeCost&Prune(N1);
minCost2 := computeCost&Prune(N2);
minCostN := min{C(S)+1,
        CSplit(N)+1+minCost1+minCost2};
if minCostN = C(S)+1
  prune child nodes N1 and N2 from tree
return minCostN
```

그림 3. 분류알고리즘의 가지치기단계
Fig. 3 Pruning Phase of Classification Algorithm

그림 3에서 N은 분류모형 내 임의의 노드이고 S는 N에 대응하는 레코드집합이다. N이 잎노드라면 N에 대응하는 최소비용서브트리는 N 자체이고 그 비용은 C(S)+1(여기서 1비트는 N이 잎노드라는 것을 나타내기 위하여 필요하다)이 된다. 반면, N이 자식노드 N_1과 N_2를 포함하는 내부노드라면 두 가지의 경우 중에서 더 적은 비용이 소요되는 경우가 선택된다. 하나의 경우는 자식노드들이 가지치기되고 노드 N은 잎노드로 처리되는 경우이고 다른 하나는 자식노드 N_1(이때 노드 N1이 루트가 되는 최소비용 서브트리여야 함)과 N_2(이때 노드 N_2가 루트가 되는 최소비용 서브트리여야 함)와 함께 노드 N이 내부노드로 처리되는 경우이다. 첫 번째 경우의 비용은 C(S)+1이 되고, 두 번째 경우의 비용은 $C_{Split}(N)+1+minCost_1+ minCost_2$ 이 된다. 결국 첫 번째 경우의 비용이 두 번째 경우의 비용을 초과하지 않으면 노드 N의 자식노드들 N_1과 N_2는 가지치기된다.

2. 관련연구 고찰 및 분석

앞에서 살펴보았듯이 분류알고리즘은 모형 생성단계와 가지치기 단계를 거쳐서 최종적으로 분류모형을 생성한다. 최종적인 분류모형을 생성하는 과정에서 보다 적은 컴퓨터자원을 사용하여 보다 빨리 분류모형을 생성할 수 있으면 효율적인 알고리즘이 된다. 또한 생성된 분류모형이 응용영역에 적합한 유의미한 분류규칙들을 많이 포함한다면 효과적인 분류 알고리즘이라고 할 수 있다.

의사결정나무 기반의 분류알고리즘과 관련하여 많은 연구들이 이루어져 왔다. SPRINT라는 분류알고리즘은 시스템효율성 향상을 주목적으로 하는 분류기법이다[8]. SPRINT는 대용량의 데이터를 처리하기 위해서 속성리스트(attribute list)라는 자료구조를 이용하여 요구되는 주기억장치 용량크기에 대한 제약을 없애고 처리할 데이터의 용량에 유연하게 대처할 수 있는 확장 가능한(scalable) 분류알고리즘이다. 또한 대용량 데이터의 신속한 처리를 위하여 병렬처리기법도 제안하고 있다. [10]에서는 시스템효율성 향상을 목적으로 AVC-group이라는 자료구조를 이용하여 SPRINT보다 적은 주기억장치 크기로 분류모형을 생성할 수 있는 기법을 제안하고 있다. 특히, 이 기법은 기존의 다양한 분류알고리즘들에 대하여 그 특성을 유지하면서 확장가능하게(scalable) 하는 일반화된 프레임워크라는 측면이 더 의미가 있다고 할 수 있다. [13]에서는 단지 2번의 데이터베이스 접근으로 분류모형을 생성하는 기법을 제안하고 있다. 이 연구 역시 분류모형의 생성시간을 최소화 하려는 시스템 효율성 중심의 논문이다.

지금까지의 연구들은 모형생성단계에서만 적용될 수 있는 기법이고 따라서, 생성된 분류모형의 과잉맞춤을 제거하기 위한 별도의 가지치기(pruning) 과정을 필요로 한다. 그러나, [9]에서는 모형생성 단계와 가지치기 단계를 통합하여 불필요한 과잉맞춤을 피함으로써 모형 생성시간을 단축시킬 수 있는 기법인 PUBLIC 알고리즘을 제안하고 있다. 분류모형을 생성하는 단계에서 서브트리 생성 비용(cost of subtree)의 하한치(lower bound)를 예측하여 가지치기(pruning) 여부를 생성단계(Building Phase)에서 미리 결정하고 불필요한 서브트리의 확장을 방지함으로써 보다 신속한 분류모형의 생성을 가능하게 한다. [14]에서는 PUBLIC 알고리즘을 확장한 것으로써 PUBLIC 알고리즘에 파라미터 제약을 적용하여 보다 단순한 분류모형을 생성하는 기법을 제안하고 있다. 사용자가 분류모형의 크기와 예측정확도를 파라미터 형태로 제시하면 그러한 제약사항에 가장 적합한 단순한 분류모형을 생성해 준다.

앞에서 살펴 본 분류알고리즘들 중에서 모형 생성단계에서의 기술적 조작을 통하여 시스템 효율을 향상시키는기법들[8,10,13]보다는, 가지치기 단계와의 통합을 통하여 시스템 효율을 향상시키려는 기법[9, 14]이 더 일반화될 수 있다는 관점에서 PUBLIC 알고리즘이 더 우수하다고 할 수 있다.

그러나 PUBLIC 알고리즘은 생성되는 분류모형의 효과성을 고려하지 않고 시스템 효율성만을 목적으로 한다는 측면에서 개선의 여지가 있다. 즉, 최종적으로 생성된 분류모형이 응용분야에 적합한 유의미한 분류규칙들을 보다 많이 포함하여야 하는데 그렇지 못한 경우가 많다는 것이다. 왜냐하면 PUBLIC 알고리즘은 서브트리 생성을 위한 분리기준(splitting criteria) 선택 시, 다른 분류 알고리즘들처럼 분리노드의 엔트로피값을 최소화 또는 최대화하는 속성과 속성값을 분리기준으로 선택하기 때문이다. 특정 그룹의 레코드들을 2개 이상의 서브 그룹들로 분리할 때 동일 서브그룹에 속한 레코드들이 동일부류(class)에 속하는 경향이 클수록 그 특정그룹의 엔트로피값은 커진다[9]. 정보전달 효율척도인 엔트로피값의 크기에 따른 분리기준은 분리되는 서브그룹의 크기를 고려하지 않는 물리적 기준으로써 과잉맞춤 현상에 의한 무의미한 분류규칙을 생성하는 근본 원인이 된다. 가지치기 단계에서 과잉맞춤 현상을 제거한다 할지라도 암호화용량을 측정하는 MDL원리에 의한 가지치기는 여전히 응용영역의 요구사항을 반영하지 않는 물리적 처리방법인 것이다. 서브그룹의 크기가 너무 작으면 훈련데이터에 의존적인 분류모형이 되어 현실세계의 실제 상황에 적용하기에는 부적합할 수 있다.

PUBLIC 알고리즘은 효율성 측면에서도 개선의 여지가 있다. PUBLIC 이전의 분류알고리즘들은 모형생성단계 후에 가지치기 단계가 수행되지만 단 한번의 가지치기 과정만 수행하면 되었다. 그러나 PUBLIC 알고리즘은 모형생성 단계의 중간에 여러 번의 가지치기 과정을 수행하는 구조로 되어 있어 단 한번의 가지치기 과정만을 수행하는 구조에 비하여 비효율적인 측면이 있다.

PUBLIC 알고리즘을 개선한 형태인 [14]의 연구에서도 이러한 문제점들은 해결되지 않았다. [14]에서도 여전히 엔트로피값을 기준으로 분리속성과 속성값을 선택하며, 모형생성단계에서의 가지치기 과정을 여러 번 수행한다. 분류모형을 단순화하기 위한 파라미터 제약이 있다 할지라도, 분류모형의 크기는 응용영역의 요구사항과 관련이 없고, 예측정확도 제약 또한 세그먼트 크기가 적합한 유의미한 분류규칙을 생성할 수 있게 하지 않는다.

3. 도전과제

생성된 분류모형에 포함된 분류규칙들이 현실세계의 실제 상황에서 유의미하게 활용되기 위해서는 일정 이상의 최소지지도와 최소신뢰도를 동시에 만족시킬 필요가 있다. 생성된 분류규칙이 훈련데이터 내에서는 100%의 신뢰도를 보인다 할지라도 10000개의 레코드 중에서 단지 2개의 레코드들로 한정된다면 단지 훈련데이터에 의존적인 분류규칙이며 현실상황에서는 무의미할 가능성이 크다. 생성된 분류규칙을 훈련데이터 내의 보다 많은 레코드들이 지지할수록 그 분류규칙은 현실상황에서 유의미할 확률이 높아질 것이다. 즉 생성된 분류규칙이 훈련데이터 내에서의 최소지지도를 만족하지 못하면 현실세계의 상황에서 유의하지 않은 규칙이 될 가능성이 크다. [Minos,2000]의 연구에서는 예측정확도 제약을 통한 최소신뢰도의 개념은 도입했으나 최소지지도의 개념은 적용하지 않았다.

본 논문에서는 효과적인 분류모형을 효율적으로 생성하기 위하여 최소지지도와 최소신뢰도의 제약을 분류알고리즘에 적용하고자 한다. 그러나 모형생성단계에서 내부노드의 확장여부는 최소지지도 제약에 의하여 결정될 것이기 때문에 내부노드의 분리기준을 단순히 엔트로피 값에 의존하여 결정할 경우 유효성이 잠재된 내부노드들을 생성할 가능성이 작아진다. 왜냐하면 엔트로피 값은 신뢰도의 개념은 포함하고 있지만 지지도의 개념은 고려하지 않기 때문에 분리된 자식노드들이 최소지지도 제약에 의하여 삭제될 수 있기 때문이다. 따라서 최소지지도를 고려하여 분류모형을 생성하고자 할 경우에는 지지도의 개념을 포함하는 새로운 분리기준 선택기법이 필요하다.

Ⅲ. 새로운 알고리즘의 제안

본 절에서는 신뢰도와 지지도의 개념을 포함하는 새로운 분리기준을 제안하며 이 분리기준에 의하여 생성된 내부노드들이 어떠한 성질들을 가질 수 있는지도 살펴본다. 그리고 새로운 분리기준에 기반한 새로운 분류알고리즘을 제안한다.

1. 새로운 분리기준

앞에서 살펴보았듯이, 기존의 엔트로피 값에 의한 분리기준은 신뢰도의 개념만을 포함하기 때문에 분리되는 새로운 내부노드나 잎노드들은 신뢰도가 높아지는 경향은 있으나 지지도에 대한 제어는 가능하지 않았다. 따라서 최소지지도 제약에 대응하기 위한 새로운 분리기준은 잎노드들의 지지도 경향을 가중치 값에 의하여 완만하게 제어할 수 있는 것이 바람직하다. 그림4는 레코드집합 S을 임의의 속성 A와 속성값 v에 의하여 S_1과 S_2로 분리하기 위한 분리기준함수 $S_{criteria}()$와 가중치적용함수 $E_w()$를 나타내고 있다.

$$S_{criteria}(S, A, v) = maximize(E_w(S_1) + E_w(S_2))$$

$$E_w(R) = W\times \sum_j \left(\frac{n_j}{n_p}\right)^2 + (1-W)\times \sum_j \left(\frac{n_j}{n}\right)^2$$

(단, W: 가중치, $0 \le W \le 1$,
n: R의 레코드 갯수
nj : R의 레코드 중 부류j에 속하는 레코드 개수
np : R의 부모노드의 레코드들의 개수)

그림 4 새로운 분리기준
Fig. 4 Novel Splitting Criteria

그림 4에서 분리기준함수 $S_{criteria}(S, A, v)$는 레코드집합 S를 S_1과 S_2로 분리 할 때 가중치적용함수 $E_w()$를 사용하면서 $E_w(S_1) + E_w(S_2)$를 최대화시킬 수 있는 속성 A와 속성값 v를 선택한다. 가중치적용함수 $E_w(R)$에서 $\left(\frac{n_j}{n_p}\right)^2$은 새로 생성되는 노드의 지지도를 고려하기 위한 부분이고 $\left(\frac{n_j}{n}\right)^2$는 신뢰도를 고려하기 위한 부분이다. 따라서 W가 커질수록 지지도의 반영비율이 커져서 최소지지도 제약에 의하여 자식노드가 삭제될 가능성은 적지만 신뢰도에 대한 반영비율은 작아지므로 신뢰도는 나빠질 수 있다. 따라서 최소지지도가 클 경우 가중치가 클수록 바람직할 수 있지만 최소신뢰도가 큰 경우는 가중치가 작을수록 좋다고 할 수 있다.

2. 새로운 분리기준의 성질

앞에서 살펴 본 새로운 분리기준 $S_{criteria}()$에 의하여 생성되는 노드들은 몇몇 특성을 갖는다. 이러한 특성들은 분류 알고리즘을 보다 효율적이고 효과적일 수 있게 하는 밑바탕이 된다. 이러한 특성들은 다음과 같다.

[정의1] S(N)은 분류모형 내 임의의 노드 N에 대응하는 레코드집합의 크기를 의미한다. □

[특성1] 분류모형 내 임의의 노드 N이 $S_{criteria}()$함수에 의하여 자식노드 N_1과 N_2로 분리될 때 $S(N) \le P$ 이면 $S(N_1) \le P$ 이고 $S(N_2) \le P$ 이다(단, P는 임의의 레코드 수). □

[특성1]에 의하여, 임의의 노드 N의 지지도가 최소지지도에 못 미칠 때, 분류모형 생성단계의 효율화를 위하여 나중에 생성될 노드 N을 루트로 하는 서브트리의 노드들을 고려할 필요없이 노드 N을 더 이상 확장하지 않는 정책이 가능하다.

[정리1] 가중치적용함수 $E_w()$의 W값이 커질수록 노드 N의 자식노드의 크기는 커진다.

(증명) 노드N의 분리기준 T_A와 T_B가 있고, T_A에 의해서는 자식노드 $N_{Ai}(i=1,...,n)$가 생성되고 T_B에 의해서는 자식노드 $N_{Bi}(i=1,...,n)$가 생성되며, $S(N_{Ai}) < S(N_{Bi})$이라고 가정하자. 또한, N_{Ai}^j는 N_{Ai}에서 부류j에 속하는 레코드들이고 N_{Bi}^j는 N_{Bi}에서 부류j에 속하는 레코드들이라고 가정하자. 분리기준함수 $S_{criteria}()$에 의하면 자식

노드들에 대한 가중치적용함수 $E_w()$값을 최대로 하는 분리기준이 선택되어야 한다. 따라서 W값이 커짐에 따라 분리기준 T_B가 선택되어 크기가 더 큰 자식노드 N_{Bi}가 생성됨을 보이면 된다.

case1) $\frac{S(N_{Ai}^j)}{S(N_{Ai})} \le \frac{S(N_{Bi}^j)}{S(N_{Bi})}$일때, $S(N_{Ai}) < S(N_{Bi})$에 의해 $S(N_{Bi}^j) \ge S(N_{Ai}^j)$가 되므로 $\left(\frac{S(N_{Ai}^j)}{S(N)}\right)^2 \le \left(\frac{S(N_{Bi}^j)}{S(N)}\right)^2$가 된다. 따라서, W값에 상관없이 T_B가 분리기준으로 선택되고 크기가 더 큰 N_{Bi}가 자식노드로 생성된다.

case2) $\frac{S(N_{Ai}^j)}{S(N_{Ai})} > \frac{S(N_{Bi}^j)}{S(N_{Bi})}$일때,

i) $S(N_{Bi}^j) \ge S(N_{Ai}^j)$인 경우 $\left(\frac{S(N_{Ai}^j)}{S(N)}\right)^2 \le \left(\frac{S(N_{Bi}^j)}{S(N)}\right)^2$가 된다. 이때 W값이 작으면 T_A가 분리기준으로 선택될 수 있지만 W값이 크면 T_B가 분리기준으로 선택되어 크기가 더 큰 N_{Bi}가 자식노드로 생성된다. ii)$S(N_{Bi}^j) < S(N_{Ai}^j)$인 경우는 N_{Bi}가 선택된다 할지라도 신뢰도가 너무 작아 의미없는 내부노드가 될 수 있으므로 무시할 수 있다. □

[정리1]에 의하여 가중치값 W가 커질수록 자식노드도 커지므로 최소지지도가 클 경우 가중치를 크게 하는 것이 바람직하다. 그러나 가중치를 크게 함에 따라 자식노드의 신뢰도를 떨어뜨릴 수 있으므로 적정한 수준의 가중치 값을 설정하는 것이 바람직하다.

[보조정리1] 가중치적용함수 $E_w()$의 W값이 작을수록 노드 N의 자식노드의 신뢰도는 커진다.

(증명) [정리1]의 경우와 비슷한 맥락에서 증명할 수 있다. 노드N에 대한 분리기준 T_A와 T_B가 있고, T_A에 의해서는 자식노드 N_{Ai}(i=1,...,n)가 생성되고 TB에 의해서는 자식노드 N_{Bi}(i=1,...,n)가 생성되며, $S(N_{Ai}) < S(N_{Bi})$이라고 가정하자. 또한, N_{Ai}^j는 N_{Ai}에서 부류j에 속하는 레코드들이고 N_{Bi}^j는 N_{Bi}에서 부류j에 속하는 레코드들이라고 가정하자.

case1) $\frac{S(N_{Ai}^j)}{S(N_{Ai})} \le \frac{S(N_{Bi}^j)}{S(N_{Bi})}$일때, $S(N_{Ai}) < S(N_{Bi})$에 의해 $S(N_{Bi}^j) \ge S(N_{Ai}^j)$가 되므로 $\left(\frac{S(N_{Ai}^j)}{S(N)}\right)^2 \le \left(\frac{S(N_{Bi}^j)}{S(N)}\right)^2$가 된다. 따라서, W값에 상관없이 T_B가 분리기준으로 선택되고 신뢰도가 더 큰 N_{Bi}가 자식노드로 생성된다.

case2) $\frac{S(N_{Ai}^j)}{S(N_{Ai})} > \frac{S(N_{Bi}^j)}{S(N_{Bi})}$일때, i) $S(N_{Bi}^j) \ge S(N_{Ai}^j)$인 경우 $\left(\frac{S(N_{Ai}^j)}{S(N)}\right)^2 \le \left(\frac{S(N_{Bi}^j)}{S(N)}\right)^2$가 된다. 이때 W값이 작으면 TA가 분리기준으로 선택되고 신뢰도가 더 큰 N_{Bi}가 자식노드로 생성된다. ii)$S(N_{Bi}^j) < S(N_{Ai}^j)$인 경우는 W값에 상관없이 T_A가 분리기준으로 선택되고 신뢰도가 더 큰 N_{Ai}가 자식노드로 생성된다.□

[보조정리1]에 의하여 가중치값 W가 작을수록 자식노드의 신뢰도가 커지므로 최소신뢰도가 클 경우 가중치를 작게 하는 것이 바람직할 것이다. 그러나 가중치를 너무 작게 함에 따라 자식노드의 지지도를 떨어뜨릴 수 있으므로 적정한 수준의 가중치 값을 설정하는 것이 바람직하다.

3. 새로운 분류알고리즘

앞에서 제안한 새로운 분리기준방법과 특성들을 이용하여 효율적이면서 효과적인 새로운 분류알고리즘을 고안할 수 있다. 그림 5는 새로운 분류알고리즘을 나타내고 있다. 그림 5에서 (1)부터 (9)사이의 처리는 의사결정나무모형을 생성하는 단계이고 (10)은 의사결정나무를 탐색하면서 최종적인 분류규칙을 생성하는 단계이다.

procedure Build&PruneTree(N, W, M_s, M_c)
/ R: 임의의 노드N에 대응하는 레코드집합,*
W : 가중치,
M_s: 최소지지도,
*M_c : 최소신뢰도 */*
(1) Initialize Root Node using data set R
(2) Initialize Queue Q to contain Root Node
(3) While Q is not empty do{
(4) Dequeue the first node N in Q
(5) for each attribute A in node N
(6) Evaluate Scriteria(R, A, v)
(7) Use best split to split node N into N_1 and N_2
(8) if $\frac{S(N_1)}{S(N)} \geq$ Ms then append N_1 to Q
(9) if $\frac{S(N_2)}{S(N)} \geq$ Ms then append N_2 to Q
*} /*end of while */*
(10) Generate Classification Rules by searching Decision Tree with checking Mc
(11) }

그림5. 새로운 분류알고리즘
Fig.5 Novel Classification Algorithm

그림 5의 (8), (9)에서 볼 수 있는 것처럼 사용자에 의해서 설정된 최소지지도 Ms는 모형생성단계에서 자식노드의 추가적인 확장여부 즉 가지치기 여부를 결정하기 위하여 사용되고 있다. PUBLIC에서는 임의의 노드의 가지치기 여부에 대하여 가지치기함수를 호출할 때마다 평가해봐야 하지만 여기서는 최소지지도와의 비교를 통한 단 한번의 평가로 충분하다. 가중치 W는 (6)에서 분리기준을 선택하기 위한 분리기준함수를 실행할 때 활용된다.

Ⅳ. 분석평가

1. 분석평가 환경

제안한 새로운 분류 알고리즘(Proposed Algorithm)의 성능을 평가분석하기 위하여 기존의 분류 알고리즘 중 성능이 좋다고 알려진 PUBLIC과 제안한 새로운 알고리즘을 C언어로 구현하였다. 실험용 데이터는 합성데이터(synthetic data)를 사용하였다. 합성데이터는 IBM Quest 사이트(www.almaden.ibm.com/cs/projects/iis/hdb/Projects/data_mining /datasets/syndata.html)에서 사용가능한(available) 데이터 생성기(Data Generator)를 사용하였다. 생성된 합성데이터의 구조는 9개의 속성들과 1개의 분류속성(2개의 속성값을 가짐)을 갖는 관계형 테이블이다. 속성들 중에서 '교육'과 '자동차', '우편번호', '신용도' 는 범주형(categorical) 데이터이고 나머지는 숫자형(numeric) 데이터이다. <표1>은 그 테이블의 속성들에 대한 설명을 나타내고 있다.

표1. 합성데이터의 레코드 속성
Table1. Record Attribute of Synthetic Data

속성	설명	값
봉급	봉급	600,000~4,500,000사이의 균일한 분포 (uniformly distributed)
커미션	커미션	if 봉급 ≥2,100,000 then 커미션은 0 else 30,000 ~ 225,000
나이	나이	20 ~ 80사이의 균일한 분포
교육	교육수준	0 ~ 4사이의 균일한 분포
자동차	자동차 보유대수	1 ~ 10사이의 균일한 분포
우편번호	주소지의 우편번호	9개의 우편번호로부터 균일하게 선택
주택가격	소유 주택의 가격	0.5 · k · 100000~1.5 · k · 100000(k∈{0,1,...,9}, 0~9사이의 구분을 우편번호에 의존적임)사이의 균일한 분포
주택기간	주택을 소유하고 있는 기간	1 ~ 30사이의 균일한 분포
대출금액	지금까지의 대출총액	0 ~ 500000사이의 균일한 분포
신용도	분류속성	예 또는 아니오

생성된 테이블들은 다양한 크기를 가지며 그 테이블들의 레코드 개수 분포는 5,000부터 50,000까지이다.

구현된 알고리즘들은 합성데이터들과 함께 1.73 GHz의 CPU와 1G 메모리를 장착한 윈도우시스템 환경에서 수행되었다.

2. PUBLIC과 Proposed의 성능 비교

제안한 알고리즘(이후부터는 'Proposed'라고 부른다)이 얼마나 신속하게 필요한 분류규칙을 생성하는지 살펴보기 위하여 기존의 알고리즘 PUBLIC과 새로운 알고리즘 Proposed의 실행시간을 비교하였다.

그림6에서 볼 수 있는 것처럼 PUBLIC과 Proposed의 최소신뢰도는 0.9로 설정하였고 Proposed의 최소지지도는 0.001로 설정하였다. 그림6에서 X축은 테이블 크기 즉 레코드들의 갯수를 나타내는 것으로써 그 단위는 5000개이다. 즉 1은 5000개의 레코드를 의미하고 2는 10000개의 레코드를 의미한다.

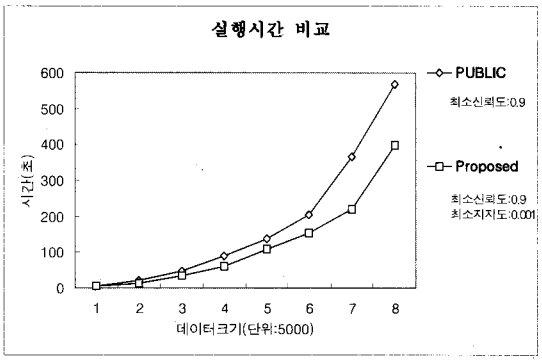


그림 6. PUBLIC과 Proposed의 실행시간 비교
Fig. 6 Comparison of Execution Time between PUBLIC and Proposed

예견한 바와 같이, Proposed가 더 신속하게 분류규칙을 생성하고 있음을 알 수 있다. PUBLIC은 트리생성과정의 중간단계 동안에 만들어지는 부분트리(partial tree, 미완성트리)들의 과잉맞춤을 특정시간 간격으로 반복해서 여러 번 가지치기(Pruning)를 해주어야 하는 반면 Proposed는 트리생성과정 동안 과잉맞춤을 하지 않기 때문에 가지치기를 할 필요가 없다. 이는 Proposed의 트리생성시간을 단축시키는 1차적인 요인이 되는 것으로 사료된다.

Proposed의 트리생성시간을 단축시키는 2차적인 요인은 불필요한 규칙들의 생성여부이다. 동일한 데이터에 대해서 PUBLIC이 Proposed보다 더 많은 규칙들을 생성하지만 PUBLIC이 생성한 규칙들 중 상당부분은 세그먼트크기가 1이거나 매우 작은 불필요한 규칙들이 많다는 것이다.

그림 7은 PUBLIC과 Proposed가 생성한 규칙들 중 지지도가 0.002이상인 규칙들의 수를 비교하고 있다. Proposed가 생성한 규칙들이 더 많음을 알 수 있다. 지지도가 큰 규칙일수록 마케팅 응용에 활용될 가능성이 큰 의미있는 규칙이라는 것을 앞에서 살펴본 바 있다.

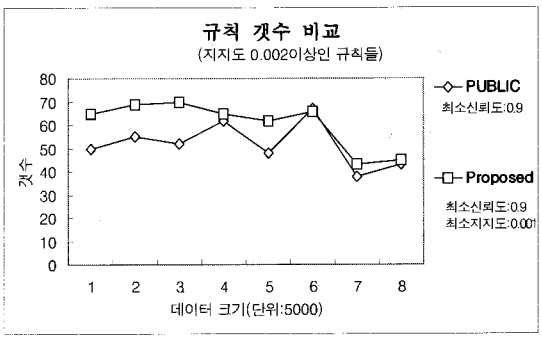


그림 7. PUBLIC과 Proposed의 유의미한 규칙 수 비교
Fig.7 Comparison of the numbers of significant rules between PUBLIC and Proposed

3. Proposed의 가중치 영향도 분석

새롭게 제안한 알고리즘은 가중치를 이용하여 분리되는 노드의 지지도와 신뢰도를 조절함으로서 보다 유의미한 분류규칙들을 생성할 수 있다. 그러나 가중치가 무조건 높다고 좋은 것이 아니라 유도하려는 분류규칙의 특성에 따라 적절한 가중치를 선택하는 것이 바람직하다.

일반적으로 가중치를 높게 설정(0.2이상 0.3 이하)하면 분리되는 노드들의 지지도가 높아지도록 분리기준이 결정되기 때문에 분류규칙의 최소지지도가 클 경우에 바람직할 수 있다. 반대로 가중치를 낮게 설정(0.1 이하)하면 분리되는 노드들의 신뢰도가 높아지도록 분리기준이 결정되기 때문에 분류규칙의 최소신뢰도가 클 경우에 바람직할 수 있다.

그림 8은 레코드 개수가 10000개이고 최소신뢰도가 0.7인 데이터 상에서 가중치와 최소지지도와의 관계를 나타내고 있다. 그림 8에서 최소지지도가 비교적 큰 경

우(0.2와 0.3)에는 가중치가 커질수록 생성되는 분류규칙 수도 많아지는 것을 볼 수 있다. 그러나 분석결과에 의하면 최소지지도가 크다 할 지라도 가중치가 0.5 이상이면 분류규칙 수가 급격히 줄어들었다. 왜냐하면 가중치가 0.5 이상이면 신뢰도에 대한 고려가 미흡한 분리기준이 선택되고 이 분류기준에 의하여 생성된 많은 분류규칙들은 최소신뢰도 제약에 못 미쳐 결국 선택되지 않기 때문이다.

그림 8에서 최소지지도가 비교적 작은 경우(0.05) 가중치가 커질수록(0.1 이상) 생성되는 분류규칙 수가 줄어드는 것을 볼 수 있다. 이는 최소지지도가 작으면 생성되는 분류규칙들 중 최소지지도를 만족시키는 분류규칙들은 많아지는데 비하여 가중치가 커짐에 따라 상대적으로 최소신뢰도 제약에 못 미치는 분류규칙들도 많아지기 때문이다.

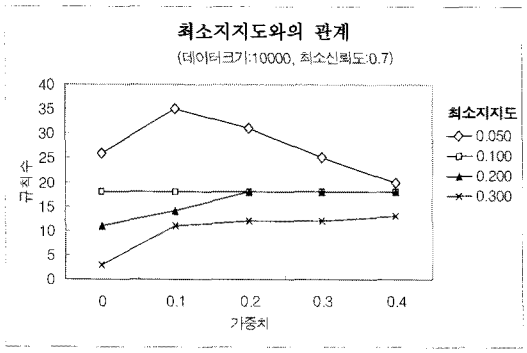


그림 8. 가중치와 최소지지도와의 관계
Fig.8 Relation between Weight and Minimum Support

결과적으로 최소지지도가 크면(0.2이상 0.5 이하) 가중치가 커질수록 생성되는 분류규칙들은 늘어나지만, 최소지지도가 작으면(0.1이하) 가중치가 커질수록 생성되는 분류규칙 수는 적어진다고 할 수 있다.

그림 9는 레코드 개수가 10000개이고 최소지지도가 0.3인 데이터 상에서 가중치와 최소신뢰도와의 관계를 나타내고 있다. 그림9에서 최소신뢰도가 비교적 큰 경우(0.900이상)에는 가중치가 커질수록(0.1이상) 생성되는 분류규칙 수가 급격하게 줄어드는 것을 볼 수 있다. 이는 앞에서 언급하였던 것처럼 가중치가 크면 분리기준에 의하여 생성된 분류규칙들의 지지도는 커질 수 있지만 신뢰도는 작아져서 최소신뢰도 제약에 못 미치게 되기 때문이다. 반면에 최소지지도가 비교적 크고(0.3) 최소신뢰도가 비교적 작은 경우(0.600)에는 가중치가 크다 할 지라도 생성되는 분류규칙들은 줄어들고 있지 않음을 알 수 있다.

결국 최소신뢰도가 커지면 가중치는 작을수록 좋다고 할 수 있다. 왜냐하면 가중치를 크게 하는 것은 신뢰도를 희생하면서 지지도가 커지도록 분리기준을 선택하기 때문이다.

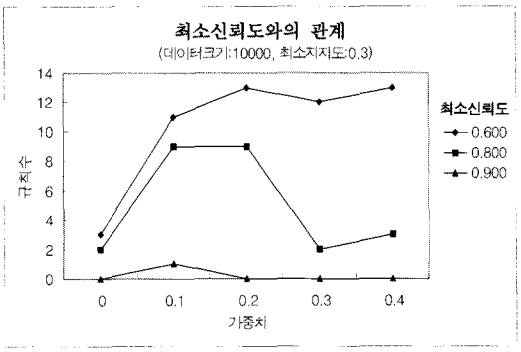


그림 9. 가중치와 최소신뢰도와의 관계
Fig. 9 Relation between Weight and Minimum Confidence

V. 결 론

ERP(Enterprise Resource Planning)와 같은 효율성 중심의 기업용 솔루션들이 비즈니스 프로세스를 혁신하고 비용을 절감하고 조직의 효율성을 꾀하였다면, 효과성을 중시하는 BI(Business Intelligence) 개념의 기업용 솔루션들을 통하여 경영의사결정을 효과적으로 지원하고 궁극적인 수익창출을 도모할 수 있다.

본 논문에서는 BI의 일종인 데이터마이닝 분야에서 효율성과 효과성을 지원하는 새로운 분류알고리즘을 제안하였다. 새로운 분류알고리즘의 이론적 기반으로서 가중치적용함수와 그 성질, 그리고 이를 바탕으로 한 새로운 분리기준 함수를 제안하였다.

제안한 분류알고리즘을 C언어로 구현하여 성능분석을 해 본 결과 다음과 같은 결과들을 얻을 수 있었다.

첫째, 제안한 분류알고리즘은 별도의 과잉맞춤 처리가 필요하지 않고 지지도가 너무 작은 불필요한 규칙들도 생성하지 않기 때문에 실행시간이 빨랐다. 이는 효율성을 더 높인 효율성 측면의 성과라고 할 수 있다. 둘째, 제안한 분류 알고리즘은 지지도가 일정 이상인 유의미한 규칙들을 기존의 방법에 비하여 보다 많이 생성하였다. 이는 분류모형의 효과성을 더 높인 효과성 측면의 성과라고 할 수 있다. 셋째, 특정 응용분야에서 요구하는 적합한 신뢰도와 지지도의 분류규칙을 유도하기 위해서는 적절한 가중치를 설정하는 것이 필요함을 알 수 있었다. 성능분석 결과에 의하면, 최소지지도가 높을 경우에는 가중치가 클수록 좋으나 최소지지도가 낮을 경우는 가중치가 클수록 오히려 생성된 규칙수가 줄어들었다. 이는 최소지지도가 작으면 생성되는 분류규칙들 중 최소지지도를 만족시키는 분류규칙들은 많아지는데 비하여 가중치가 커짐에 따라 상대적으로 최소신뢰도 제약에 못 미치는 분류규칙들도 많아지기 때문이다. 또한, 최소신뢰도가 커지면 가중치는 작을수록 좋다고 할 수 있다. 왜냐하면 가중치를 크게 하는 것은 신뢰도를 희생하면서 지지도가 커지도록 분리기준을 선택하기 때문에 신뢰도가 떨어질 수 있기 때문이다.

본 연구의 결과는 효과성을 중시하는 경영정보시스템인 비즈니스인텔리전스의 기반기술로써 활용될 수 있을 뿐만 아니라 온톨로지 기반 정보시스템 구축에서도 중요한 역할을 할 수 있다.

본 논문에서는 적절한 가중치 값에 대한 체계적인 접근이 이루어지지 않았다. 이 논문을 기점으로 향후, 응용분야에 적합한 적절한 가중치에 대한 연구가 필요하다.

참고문헌

[1] 공유근, 데이터마이닝 기법들을 이용한 온톨로지 생성, 고려대 대학원 석사학위논문, 2004.

[2] Victor S.Y.Lo, "The True Lift Model-A Novel Data Mining Approach to Response Modeling in Database Marketing", ACM SIGKDD Explorations Newsletter, Volume 4 Issue 2, 2002, pp.78-86.

[3] R.Agrawal, T. Imielinski, and A.Swami, "Mining Association Rules between Sets of Items Large Databases", In Proceedings of ACM SIGMOD Conference om Management of Data, Washington D.C., May, 1993, pp.207-216.

[4] R.Agrawal and R.Srikant, "Fast Algorithms for Mining Association Rules", In Proceedings of the 20th VLDB Conference, Santiago, Chile, Sept., 1994, pp.487-499.

[5] Bing Liu, Wynne Hsu and Yiming ma, "Mining Association Rules with Multiple Minimum Supports", In Proceedings of ACM SIGKDD(KDD-9),1999 pp.337-341..

[6] 하단심, 황부현, "의미있는 희소 데이터를 포함한 연관규칙탐사기법", 정보과학회논문지, 2001.

[7] 김근형, 황병웅, 김민철, "중요지지도를 고려한 연관규칙 탐사 알고리즘", 정보처리학회논문지 D, 제11-D권 제 3호, 2004, pp.545-552..

[8] John Shafer, Rakesh Agrawal and Manish Mehta, "SPRINT:A Scalable Parallel Classifier for Data Mining", In Proceedings of the 22nd VLDB Conference, India, 1996, pp.1-12..

[9] Rajeev Rastigi, Kyuseok Shim,"PUBLIC : A Decision Tree Classifier that Integrates Building and Pruning", In Proceedings of the 24nd VLDB Conference, New York, USA, 1998, pp.404-415..

[10] Johannes Gehrke, Raghu Ramakrishnan and Venkatesh ganti, "RainForest-A Framework for fast Decision Tree Construction of large Datasets", In Proceedings of the 24th VLDB Conference, NewYork, USA, 1998, pp.416-427.

[11] 김은영, 핸드폰 시장에서의 시장세분화에 관한 연구, 국민대 대학원 석사논문, 2007.

[12] J. R. Quinlan and R. L. Rivest. Inferring decision trees using minimum description length principle. Information and Computation, 1989.

[13] Johannes Gehrke, Venkatesh Ganti and Raghu Ramakrishnan, "BOAT-Optimistic Decision Tree Construction", SIGMOD '99 Philadelphia PA, 1999,

pp.169-180.

[14] Minos Garofalakis, Dongjoon Hyun, Rajeev Rastogi and Kyuseok Shim,"Efficient Algorithms for Constructing Decision Trees with Constraints", ACM SIG-KDD 2000, Boston, USA, 2000, pp.335-339.

저자소개

김 근 형(Keun-Hyung Kim)

1990년 2월 서강대 컴퓨터학과 (공학사)
1992년 2월 서강대 컴퓨터학과 (공학석사)
2001년 2월 서강대 컴퓨터학과 (공학박사)
2001년 9월 ~ 현재: 제주대 경영정보학과 부교수
※관심분야 : 데이터마이닝, 텍스트마이닝, e-비즈니스