데이터 마이닝을 위한 개선된 직사각형 분해 알고리즘

송 지 영† · 임 영 희†† · 박 대 희†††

요 약

본 논문에서는 동적으로 변화하는 대용량의 데이터베이스로부터 보다 현실적인 데이터 마이닝의 수행을 가능케 하기 위하여 기존의 직사각형분해 알고리즘을 개선한 새로운 알고리즘을 제안한다. 새로운 알고리즘은 이진 행렬을 이분(bipartite) 그래프로 변환하고, 변환된 이분 그래프에서 이분클리크(biclique)를 찾음으로써 직사각형 분해를 수행한다. 제안된 알고리즘은 새롭게 유도된 수학적 정리들을 바탕으로 출발하였으며, 복잡도 분석을 통하여 그 효율성을 보이고, 기존의 분류 방법론과의 비교를 통하여 제안된 방법론이 규칙의 수와 분류율면에서 우수함을 보인다.

An Improved Rectangular Decomposition Algorithm for Data Mining

Ji-Young Song† · Young-Hee Im†† · Dai-Hee Park†††

ABSTRACT

In this paper, we propose a novel improved algorithm for the rectangular decomposition technique for the purpose of performing data mining from large scaled database in a dynamic environment. The proposed algorithm performs the rectangular decompositions by transforming a binary matrix to bipartite graph and finding bicliques from the transformed bipartite graph. To demonstrate its effectiveness, we compare the proposed one which is based on the newly derived mathematical properties with those of other methods with respect to the classification rate, the number of rules, and complexity analysis.

키워드 : 직사각형 분해(Rectangular Decomposition), 점증적 갱신(Incremental Updating), 데이터 마이닝(Data Mining)

1. 서 론

최근 정보기술의 발달과 인터넷이라는 새로운 패러다임의 등장으로 인하여 데이터베이스에 저장되는 데이터의 양은 폭발적으로 증가하고 있다. 또한 지식기반 사회로의 진입은 수집된 대규모의 데이터로부터 유용한 정보와 지식들을 추출할 수 있는 새로운 지능적 분석기법과 도구들을 절실히 요구하고 있는 상황이다. 최근 이러한 요구를 충족시키기 위한 방법론으로 데이터 마이닝 기술에 대한 관심이 집중되고 있다. "데이터 마이닝"이란 대량의 실제 데이터로부터, 이전에 잘 알려져 있지 않으며 묵시적이고 잠재적으로 유용한 정보를 추출하는 작업을 통칭한다[1]. 이러한 데이터 마이닝 기술은 인공지능 입장에서 바라볼 경우, 기계학습(machine learning) 분야와 매우 밀접한 관계를 갖는다. 기계 학습 방법은 복잡한 의사 결정 문제를 다루기 위한 성공적인 원칙(discipline)을 제공하며, 특히 의사 결정을 위한 분류기(classifier)의 구축을 위해 데이터베이스로부터 지식탐사(knowledge discovery)를 하는데 사용되어 왔다[2, 3]. 그러나 데이터 마이닝에서 사용되는 실세계 데이터베이스의 데이터는 기계학습에서 사용되는 데이터와는 달리, 빈번한 삽입과 삭제 그리고 갱신에 의해 동적(dynamic)으로 변화한다. 따라서 동적으로 변화하는 환경 하에서, 보다 효율적인 데이터 마이닝을 수행하려면 지식기반(knowledge-base) 또는 규칙기반(rule-base)에 대한 점증적 갱신(incremental update)이 가능한 방법들이 요구된다. 점증적 갱신 기술이란, 데이터베이스의 갱신시 이미 발견된 규칙들을 모두 무시하고 새롭게 재구성하는 기존 기술력의 한계를 극복하는 매우 의미 있는 것으로, 동적으로 변하는 데이터베이스의 특성에 대처하기 위한 방법론이다. 즉, 새로운 데이터의 추가, 삭제 또는 갱신시 전체 데이터베이스를 재구성하여 재학습하는 대신 새로운 데이터에 대한 부분학습 및 일부 상관관계를 갖는 규칙들에 대해서만 조정작업을 수행

※ 본 연구는 고려대학교 특별연구비의 지원으로 수행되었음.
† 준 회 원 : 고려대학교 대학원 전산학과
†† 정 회 원 : 대전대학교 컴퓨터정보통신공학부 교수
††† 정 회 원 : 고려대학교 컴퓨터정보학과 교수
논문접수 : 2003년 1월 17일, 심사완료 : 2003년 4월 26일

한다[3].

대부분의 귀납적 학습론(inductive learning methods)들은 초기 데이터 셋으로 학습을 종료한 후, 데이터 셋의 변화가 있을 시, 이미 새롭게 추가된 데이터에 대해서만 규칙기반을 재구성하는 대신, 처음부터 전체 데이터에 대해 다시 학습을 수행하여 규칙 기반을 갱신한다. 따라서 이러한 방법론들은 계산비용의 과다로 인하여 데이터 마이닝 관점에서 볼 때 그 실용성이 떨어진다. Khcherif[8]등은 이러한 문제점을 극복하기 위한 대안으로 점증적 갱신이 가능한 직사각형분해 기법을 제안하였다. Khcherif[8]등에 의해 제안된 직사각형 분해 기법은 규칙을 직사각형 형태로 저장하고, 새로운 데이터가 추가될 때, 추가되는 데이터와 상관 관계가 있는 직사각형(규칙)만을 재구성함으로써 규칙기반을 점증적으로 갱신한다. Maddouri등[7]은 부정확하거나 애매 모호한 현실 세계의 특성을 반영한 데이터베이스를 처리하기 위하여 퍼지(fuzzy)이론[4-6]을 직사각형 분해 기법에 도입한 방법을 제안함으로써 한 단계 발전적인 직사각형 분해 기법을 완성하였다. 그러나 두 방법[7, 8] 모두 점증적 갱신 능력을 갖고 있다는 장점에도 불구하고, 직사각형 분해 문제를 그래프로부터 클리크(clique)를 찾는 문제(즉, NP-하드 문제)로 변환하여 휴리스틱으로 풀고 있기에 현실적으로 실용성이 떨어지는 방법론이다.

따라서 본 논문에서는 동적으로 변화하는 환경 하에서 빈번한 삽입과 삭제, 그리고 갱신이 이루어질 때, 직사각형 분해기법의 최대 장점인 점증적 갱신 능력은 유지하되, 최적 커버리지를 찾는 비용을 현저히 줄이는 보다 개선된 알고리즘을 제안하고자 한다. Khcherif와 Maddouri등이 사용한 직사각형 분해 알고리즘에서는 이진 행렬을 일반(general) 그래프로 변환한 후, 변환된 일반 그래프에서 최대 클리크를 찾음(즉, NP-하드 문제[8])으로써 직사각형 분해를 수행한다. 그러나 본 논문에서는 이진 행렬을 일반그래프로 변환하지 않고, 이분 그래프로 변환하여, 변환된 이분 그래프에서 이분클리크를 찾음(노드 - 삭제(node-deletion) 방법을 이용하면 다항(polynomial) 시간 안에 풀 수 있는 문제[9, 10])으로써 직사각형 분해를 수행하는 보다 개선된 알고리즘을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 기존의 직사각형 분해기법에 관하여 간략히 살펴보고, 문제점들을 알아본다. 3장에서는 새롭게 유도된 수학적 정리들을 바탕으로 개선된 직사각형 분해 알고리즘을 제안하고, 복잡도 분석을 통해 그 효율성을 보인다. 4장에서는 실험결과를 통하여 제안된 알고리즘과 기존의 방법론들을 비교·분석함으로써 제안된 알고리즘의 효율성을 확인한다. 마지막으로 5장에서는 결론 및 향후 연구방향에 대하여 논한다.

2. 직사각형 분해 기법(Rectangular Decomposition Technique)

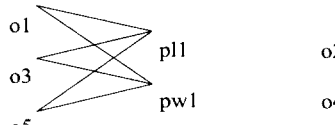
직사각형 분해 기법이란 주어진 관계형 데이터베이스를 객체(object)들의 집합 O와 특성(property) 혹은 속성값(attribute value) 들의 집합 P사이의 이진 관계 $R(O, P)$로 표현한 후, 이진 관계 R의 최적 커버리지(optimal coverage)를 찾음으로써 데이터베이스의 내재된 정보를 잃지 않으면서 여러 개의 작은 직사각형으로 분해하여 저장하는 기술이다[8]. 이때 얻어진 작은 직사각형들은 하나의 규칙으로 일대일 매핑되어 해석되며 이진 관계 R의 최적 커버리지 알고리즘에 의하여 최소의 직사각형 즉, 최소의 규칙기반이 형성된다.

이진 관계에서 최적 커버리지를 찾는 문제는 완전 부분행렬(complete sub-matrix)의 개수를 최소로 하여 커버링(covering)하는 문제로써, Maddouri등[7]은 이진 행렬로부터 얻어진 이분 그래프를 일반 그래프로 변환한 후 최대(maximum) 클리크를 찾음으로써 직사각형 분해를 수행하였다. 이는 m×n 행렬(m : 속성의 수, n : 항목의 수)을 (m+n)×(m+n) 행렬로 변환한 후 해를 찾는 것이므로, n의 개수가 m에 비해 매우 큰 대용량 데이터베이스의 경우, 위 방법은 계산비용이 매우 커지는 한계점을 가질 수밖에 없다. 다음의 예제는 퍼지 이진 관계의 직사각형 분해와 최적 커버리지로부터 얻어진 퍼지 규칙 기반을 보여준다.

[예제] 본 예제에서 사용된 데이터는 IRIS[14] 데이터의 일부로써, 각 항목들은 하나의 데이터를 의미한다. 또한 pl과 pw는 4개의 속성 중 각각 petal length와 peta l width를 의미하며, 퍼지 소속함수에 의해 각각 pl 1, pw 1(low), pl 2, pw 2(medium), pl 3, pw 3(high)로 나뉜다.

〈표 1〉 퍼지 이진 관계 R

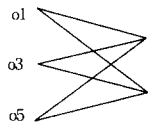
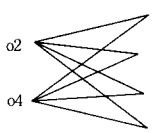
속성 / 항목	pl 1	pl 2	pl 3	pw 1	pw 2	pw 3
o 1	1	0	0	1	0	0
o 2	0	0.75	0.25	0	0.67	0.33
o 3	1	0	0	1	0	0
o 4	0	0.87	0.13	0	0.67	0.33
o 5	1	0	0	1	0	0



RE_1 = { {o1, o3, o5} × {pl1, pw1} } RE_2 = { {o2, o4} × {pl2, pl3, pw2, pw3} }
Optimal coverage of R_α (α = 0.13) : CV = { RE_1, RE_2 }

(그림 1) 퍼지 이진 관계에 대한 직사각형 분해 결과

〈표 2〉 (그림 1)의 최적 커버리지로부터 얻어진 퍼지규칙

최적 직사각형	규 칙	정확도
 RE1 은 C1 class에 속한다	**If** low petal length **AND** low petal width **THEN** 'Setosa' flower.	3/3 = 1
 RE2 은 C2 class에 속한다	**IF** (medium **OR** high petal length) **AND** (medium **OR** high petal width) **THEN** 'Versicolor' flower.	2/2 = 1

3. 개선된 알고리즘

본 논문에서는 동적으로 변화하는 환경 하에서 빈번한 삽입과 삭제, 그리고 갱신이 이루어질 때, 직사각형 분해기법의 최대 장점인 점증적 갱신 능력은 유지하되, 최적 커버리지를 찾는 비용을 현저히 줄이는 보다 개선된 알고리즘을 제안하고자 한다. 이를 위해 이진행렬을 그래프로 변환하여 얻어진 이분 그래프를 기초로 하여 다음의 정리들을 유도한다.

[정리 1] 주어진 이분 그래프를 $B=(V_1\cup V_2, E)$라하고, B의 부분집합인 완전 이분 그래프(complete bipartite graph)를 $B_c=(W_1\cup W_2, E_c)$이라 하자. 이때 $B_c=W_1\times W_2$, $W_1\subset V_1$, $W_2\subset V_2$, $E_c\subset E$이다. 만약 노드-삭제를 이용하여 구해진 직사각형을 B_c라 하면, B_c는 극대(maximal) 직사각형이다.

증명) 모순(contradiction)에 의한 증명을 이용하기 위하여 결론을 거짓이라 가정하자. 그러면 B_c는 극대 직사각형이 아니다. 즉 $B_c = W_1 \times W_2 \subset W_1' \times W_2' = B_c'$인 B_c'이 존재하게 된다. B_c가 B_c'의 진부분집합이 되는 경우는 다음의 세 경우이다.

① $W_1' = W_1 + x$ and $W_2' = W_2$

② $W_1' = W_1$ and $W_2' = W_2 + y$

③ $W_1' = W_1 + x$ and $W_2' = W_2 + y$

①의 경우를 살펴보면, V_1의 원소를 $u_1, u_2, \cdots, u_m$, V_2의 원소를 $v_1, v_2, \cdots, v_n$, W_1의 원소를 $w_1, w_2, \cdots, w_i$, W_2의 원소를 $z_1, z_2, \cdots, z_j$라하고, x는 $x \in V_1$ and $x \notin W_1$인 즉, $V_1 - W_1$의 임의의 원소라 하자. $u_k = w_k$, $v_l = z_l (1 \le k \le i, 1 \le l \le j)$로 배열하면 x는 $u_{i+1}, u_{i+2}, \cdots, u_m$중의 하나가 된다. 이때, $W_1 + x = W_1'$라 하면 $W_1' \times W_2 = B_c'$가 존재하게 된다. 여기서 B_c'는 완전 이분 그래프이므로 x는 W_2의 모든 원소와 이음선(edge)을 갖고 있어야만 한다. 그러나 노드-삭제방법에서는 그래프에서 $\forall z(\in W_2)$와 이음선을 갖고 있지 않는 노드만을 삭제한다. 따라서, $x \in V_1$ and $x \notin W_1$인 x는 존재하지 않는다. 이것은 가정 ①에 위배되므로 노드-삭제를 이용하여 구해진 직사각형 B_c는 극대 직사각형이다. ②와 ③의 경우에도 같은 방법으로 증명된다. ■

[따름정리] 주어진 이분 그래프를 $B=(V_1\cup V_2, E)$라 하고, B의 부분집합인 완전 이분 그래프를 $B_c=(W_1\cup W_2, E_c)$이라 하자. 이때 $B_c=W_1\times W_2$ $W_1\subset V_1, W_2\subset V_2, E_c\subset E$이다. 노드-삭제를 이용하여 B로부터 구한 B_c를 B_{c_1}이라하고, B_{c_1}에 포함되지 않은 이음선 즉, $e_2 \notin E_{c_1}$로부터 시작하여 구한 B_c를 B_{c_2}라 하자. 같은 방법으로 $e_3 \notin (E_{c_1}\cup E_{c_2})$로부터 시작하여 구한 B_c를 B_{c_3}라 하고, 모든 이음선 $e_i \in E$가 모두 사용될 때까지 B_{c_3}을 구하자. 여기서 구해진 B_c들의 집합을 $CV=\{B_{c_1}, B_{c_2}, B_{c_3}, \cdots, B_{c_n}\}$라 하면 CV는 커버리지이다.

증명) [정리 1]에 의해 $B_{c_i} \in CV$는 모두 극대 직사각형이고, 모든 이음선 $e_i \in E$는 CV에 포함되었음으로, 커버리지의 정의에 의해 $CV=\{B_{c_1}, B_{c_2}, B_{c_3}, \cdots, B_{c_n}\}$는 커버리지이다. ■

이때 계산비용의 절감을 위해 최적 해에 가까운 근사 해를 최적 직사각형 즉, 생성된 규칙으로 간주해도 전체 규칙 기반의 성능은 크게 영향을 받지 않는다. 따라서 본 논문에서는 노드-삭제를 이용하여 찾아진 극대 직사각형(근사해)들로 구성된 커버리지 중에서 직사각형의 수를 최소화하는 커버리지를 최적 커버리지로 사용한다.

[정리 2] 노드-삭제에 의해 구해진 극대 직사각형 B_{c_i}를 $Rec_i (i = 1, 2, \cdots, n)$라 하고, [따름정리]에 의해 구한 커버리지를 $CV_{opt} = \{Rec_1, Rec_2, \cdots, Rec_n\}$라 하자. 여기서 각각의 극대 직사각형 Rec_i를 최적 직사각형이라 놓으면, CV_{opt}는 최적 커버리지이다.

증명) 각각의 직사각형이 최적 직사각형이라고 했으므로, 정의에 의해 CV_{opt}에 있는 모든 Rec_i가 잉여 직사각형이 아님을 보이면 된다. 즉, n-1개의 Rec_i로는 커버리지를 구성하지 못함을 보이면 된다. 노드-삭제에서 $Rec_i\,(i=1,2,\cdots,n)$를 구하기 위하여 처음에 선택되는 노드의 쌍, 즉 하나의 이음선을 e_i라 하자. 먼저 Rec_n이 잉여 직사각형인지 살펴보자. 커버리지의 정의에 의해 모든 이음선은 적어도 하나의 직사각형에는 포함되어야 한다. 그러나 e_n은 Rec_1, Rec_2, …, Rec_{n-1}이 CV_{opt}에 포함된 후에도 어떤 $Rec_i\,(i=1,2,\cdots,n-1)$에도 포함되지 않았고, 각각의 $Rec_i\,(i=1,2,\cdots,n-1)$는 최적 직사각형이므로, e_n을 포함하는 최적 직사각형 Rec_n은 CV_{opt}에 반드시 포함되어야만 한다. 즉, Rec_n은 잉여 직사각형이 아니다. 같은 방법으로 $Rec_1, Rec_2, \cdots, Rec_{n-1}$도 커버리지에 반드시 포함되어야만 한다. 또한 각각의 직사각형이 최적 직사각형이므로 다른 어떤 직사각형에도 포함될 수 없고, 여러 개의 직사각형에 나뉘어서 포함될 수도 없다(나머지 직사각형들이 최적 직사각형이므로). 따라서 n-1개 직사각형으로는 커버리지를 구성할 수 없으므로 정의에 의해 CV_{opt}는 최적 커버리지이다. ■

지금까지 유도한 정리에 의해 직사각형 분해시, 이진 행렬로부터 얻어진 이분 그래프를 일반 그래프로 변환하지 않고, 이분 그래프에서 이분클리크를 찾음으로써 직사각형 분해를 수행하여도 원하는 최적 커버리지를 얻을 수 있음이 증명되었다. 이분 그래프에서 이분클리크를 찾는 문제는 찾고자하는 해가 포함되어있는 탐색공간이 m×n 행렬이므로 Maddouri등의 방법론의 (m+n)×(m+n) 행렬보다 매우 작아지게 된다. 뿐만 아니라 일반 그래프에서 최대 클리크를 찾는 문제는 NP-하드 문제이고, 이분 그래프에서 이분클리크를 찾는 문제는 다항 시간만에 풀 수 있는 문제이다. 따라서 본 논문에서 제안된 알고리즘이, NP-하드 문제를 휴리스틱을 이용하여 해결한 Maddouri등의 방법론에 비해 보다 효율적인 방법론이라 할 수 있다.

다음은 위에서 살펴본 정리들을 바탕으로 본 논문에서 제안하는 개선된 직사각형 분해 알고리즘을 보여준다. 그래프에서 도메인의 노드, 코도메인의 노드, 이음선은 각각 데이터베이스에서 속성, 항목, 그리고 관계를 나타낸다.

```
입력 : 정보 테이블(이진 관계)
출력 : 최적 커버리지(규칙기반)
Opt_Cov()
{
   Coverage = ∅ ;
   Remain_Edge = All_Edge ;
   All_List = Dom ∪ Cod ;
   Candidate = ∅ ;
   while ( Remain_Edge != ∅ )
   {
      Coverage = Coverage ∪ Max_Rec (All_List, All_Edge) ;
      Remain_Edge = Remain_Edge - edge (Max_Rec) ;
      Candidate = Candidate_Node (Remain_Edge) ;
   }
   return (Coverage) ;
}
```

(그림 2) 이분 그래프로부터 최적 커버리지를 구하는 메인 함수

```
입력 : List와 Edge
출력 : 극대 직사각형
Max_Rec (List, Edge)
  {
    Maximal = Candidate ;
    List = List - Candidate ;
    ∀n ( ∈ List)  s. t. (Candidate, n) ∉ Edge
          List = List - n ;
    while ( List != ∅ ) ;
    {
       n = Best_node (List) ;
       Maximal = Maximal ∪ {n} ;
       List = List - n ;
       ∀n'(∈ List)  s. t. (n, n') ∉ Edge
            List = List - n' ;
    }
    return (Maximal) ;
}
```

(그림 3) 극대 직사각형을 찾는 알고리즘

Candidate_Node()는 이미 커버리지에 포함된 이음선을 제외한 나머지 이음선을 입력으로 받고, 이득(gain)값의 감소를 최소화하는 두 개의 연결된 노드를 찾아 반환한다. Best_Node()는 현재 List에서 선택된 노드에 의해 직사각형 이득값의 감소를 최소화하는 노드를 반환하는 함수이다. Max_Rec은 리스트가 공집합이 될 때까지 반복문을 수행한다. Best_Node n이 결정되면 n을 Maximal에 넣고, n과 연결되지 않은 모든 노드들을 제거한다. 리스트에 있는 모든 노드들이 삭제 또는 Maximal에 포함되면(즉 리스트가 공집합이면), 현재 직사각형이 극대 직사각형이 된다. 이는 [정리 1]에서 증명한 바 있다.

Opt_Cov()는 Remain_Edge가 공집합이 될 때 정지하게 된다. 즉, 도메인과 코도메인 사이의 모든 이진 관계가 최적 커버리지에 포함될때 해를 구하게 되는 것이다. 처음 Coverage에 Max_Rec이 포함되면, Candidate_Node를 찾는다. 이것은 현재 커버리지에 포함되지 않은 이음선 중에서 다음의 최적 직사각형에 포함될 확률이 높은 이음선을 두 번째 직사각형을 찾는데 사용하기 위함이다. 이렇게 함으로써 커

버리지에 포함된 최적 직사각형의 개수를 최소화할 수 있다. [정리 2]에 의해 Opt_Cov()에서 구해진 커버리지가 최적 커버리지임을 알 수 있다.

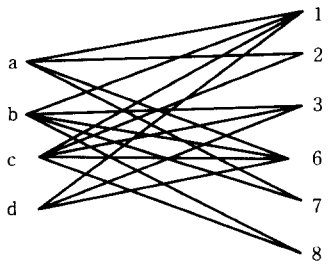
Best_Node에서 사용되는 이득함수는 $gain = |Dom| \times |Cod| - (|Dom| + |Cod|)$로 정의된 것을 사용한다. 따라서 노드의 수가 같을 때, 또는 도메인과 코도메인의 노드 개수의 차이가 작을 때(즉, 정사각형에 가까울 때), 이득 함수는 가장 큰 값을 갖는다. 위의 이득 함수를 사용한 이유는 도메인의 개수만 너무 많으면, 지지도는 작으면서 너무 구체적인(specific) 규칙이 되고, 반대로 코도메인의 개수만 너무 많으면 지지도는 크지만 규칙의 조건부인 속성의 수가 작아지므로 너무 일반적(general)인 규칙이 되기 때문이다. 따라서 해결하려는 문제의 성격에 따라 이득함수를 조정할 수 있다.

[예제] <표 3>에 나타난 이진 관계를 본 논문에서 제안한 개선된 알고리즘을 이용하여 직사각형 분해한다.

〈표 3〉 이진 관계

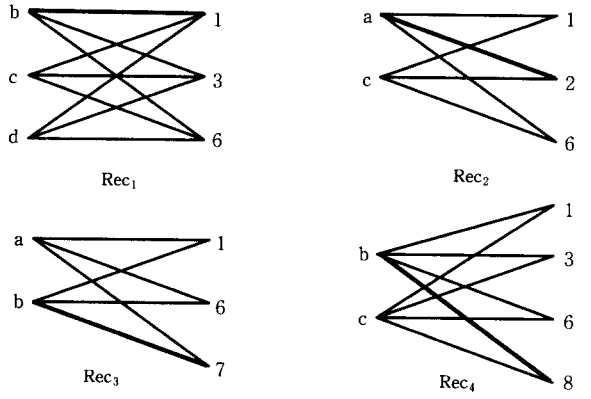
항목 \ 속성	a	b	c	d
1	1	1	1	1
2	1	0	1	0
3	0	1	1	1
6	1	1	1	1
7	1	1	0	0
8	0	1	1	0

주어진 이진 관계를 그래프로 나타내면 (그림 4)와 같다. 이것을 Opt_Cov()에 입력으로 넣으면, (그림 5)와 같이 $Rec_1, Rec_2, Rec_3, Rec_4$가 차례로 구해진다. 이때 Candidate _Node의 쌍, 즉 이음선은 각 직사각형의 굵은 선으로 표현된다.



(그림 4) 이진 관계에 대한 이분 그래프

본 논문에서 제안한 개선된 직사각형 분해 알고리즘은 주어진 이진 관계로부터 최적 커버리지를 찾는 부분만을 새롭게 수정한 것이므로, 기존에 Maddouri[7]등이 제안한 방법론의 장점인 점증적 갱신 부분은 그대로 유지할 수 있다. 다음은 본 논문에서 제안된 직사각형 분해 알고리즘(Opt_Cov())을 사용한 완성된 점증적 갱신의 직사각형분해 알고리즘이다.



(그림 5) 이진관계에 대한 최적 커버리지

```
Incremental_Update ( )
{
  Let P be partitioned to packages P1, P2, ···, Pp
  for (k = 1 ; k =< p ; k++)
  {
    R' = Pk ∪ {falsified rectangles}
    CVk = CVk − {falsified rectangles}
    CVk = CVk ∪ Opt_Cov(R')
  }
}
```

(그림 6) 점증적 갱신 직사각형 분해 알고리즘

이진 관계 R이 p개의 패키지(package)로 분할 (P_1, P_2, ···, P_p) 되었다고 하자. 각각의 패키지는 R의 부분집합이고, 하나 이상의 열을 나타낸다. 우리는 다음과 같이 단계적으로 이진관계 R에 대한 최적 커버리지를 찾는다.

첫 번째 단계에서, $R = P_1$이다.

k번째 단계에서, $R_k = P_1 \cup P_2 \cup \cdots \cup P_k$이라 하고, CV_k는 R_k의 최적 커버리지라 하자. 그러면 단지 CV_k와 P_{k+1}만을 이용하여 $R_{k+1} = R_k \cup P_{k+1}$의 최적 커버리지를 구할 수 있다.

(그림 6)의 알고리즘에서 falsified rectangles은 새로 추가된 P_k와 상관관계가 있는 직사각형들을 말한다.

동적으로 변화되는 데이터베이스에 새로운 데이터가 추가될 때, 계산될 도메인의 크기가 전체 데이터베이스가 아닌 상관관계가 있는 일부의 데이터만을 이용하여 규칙 기반을 재구성할 수 있도록 함으로써 규칙기반의 갱신 성능이 향상됨을 알 수 있다. 이러한 계산 비용 향상은 대용량의 데이터베이스에서 그 의미가 더욱 크다.

● **복잡도 분석(Complexity Analysis)**

기존의 직사각형 분해 알고리즘과 본 논문에서 제안한 개선된 직사각형 분해 알고리즘의 계산 복잡도를 비교함으로써 제안한 알고리즘의 효율성을 확인하고자 한다. Opt_Cov() 알고리즘은 하나의 이음선(Candidate_Node의 쌍)에 대해 한번의 Max_Rec()를 호출(call)한다. 호출된 Max_Rec()에서는 직사각형을 구성하기 위하여 List가 공집합이 될 때까지 노드를 선택 및 삭제한다. 따라서 전체 알고리즘의 복잡도는 커버리지에 포함되기 위하여 선택되는 노드의 개수와 동일하다.

$c = card(cod(R))$이고 $d = card(dom(R))$이라 하자. Max_Rec가 호출되는 횟수는 커버리지 내의 직사각형의 개수이므로 최악의 경우에 전체 이음선의 개수가 되고 이것은 $c \times d$이다. 또한 하나의 이음선에 대해 하나의 직사각형을 구성함으로, 하나의 직사각형을 구성하기 위해 선택되는 노드의 개수는 최악의 경우에 모든 노드의 개수인 $c + d$이다. 따라서 전체 복잡도는 이 둘을 곱한 $O(cd(c+d))$가 된다. 이것은 노드 삭제가 전혀 발생하지 않는 최악의 경우로 계산한 것이고, 실제로는 불가능한 상황이다. 일반적으로 실험에 의한 복잡도는 이보다 매우 낮은 $O(c)$로 나타난다. NP-하드 문제를 휴리스틱을 이용하여 해결한 기존의 Maddouri[7]등의 알고리즘은 최악의 경우에 $O((cd)^2(c+d)^2)$의 복잡도를 갖는다. 일반적으로 대용량의 데이터베이스에서 지식탐사에 사용되는 관계는 $d \ll c$이다. 따라서 Maddouri[7]의 방법은 $O(c^4)$임에 비해, 본 논문에서 제안한 방법은 $O(c^2)$이 된다. 따라서 본 논문에서 제안한 개선된 알고리즘은 직사각형 분해를 위한 효율적인 방법론이라 할 수 있다. 또한 기존의 기계학습 방법인 PVM, IPR, FIPR, CART, ID3, SPINA[8, 15, 16]등과의 복잡도 및 지식 표현 형태를 <표 4>에서 비교해 보면 제안된 방법론이 기존의 방법론에 비해 복잡도와 지식표현 면에서 우수함을 확인할 수 있다.

〈표 4〉 계산 복잡도와 언어 표현 비교

	방법론	복잡도	지식 표현
Rule Induction 방법	PVM	$\sum_{k=1}^{c}(c^2d)^{2^{k-1}}$	Symbolic
	IPR	$(cd)^2(c+d)^2$	Symbolic
	FIPR	$(cd)^2(c+d)^2$	Linguistic
	제안된 방법론	$cd(c+d)$	Linguistic
Decision Tree 방법	CART	$c^{11} \times d$	Tree
	ID3	$c^2 \times d^2$	Tree
	SPINA	$c^2 \times d^2$	Latticial graph

4. 실험 및 결과

본 장에서는 제안된 알고리즘의 타당성과 효율성을 검증하기 위해 IRIS 데이터를 이용하여 실험하였다. 학습 자료의 개수에 따른 성능평가를 위해 학습 데이터를 21, 30, 60, 90으로 실험하되 점증적 갱신의 효과를 확인하기 위하여 먼저 21개의 데이터로 학습한 뒤 30, 60, 90개의 데이터에 대한 학습은 여기에 각각 9개, 30개, 30개씩을 더하여 규칙기반을 점증적으로 갱신하여 실험한다. 각 실험에 대하여 1000번씩 수행 후 상·하위 각 10%씩을 제외한 평균값으로 기존의 방법론들과 비교한다.

<표 5>는 본 논문에서 제안된 알고리즘으로 테스트 데이터를 분류한 결과와 퍼지 규칙을 이용한 기존의 분류 알고리즘들의 결과에 대한 성능을 비교한 것이다. <표 5>에서 보는 바와 같이 본 논문에서 제안한 알고리즘이 기존의 방법에 비하여 규칙의 수는 적고 분류율은 높음을 알 수 있다. 또한 위 실험에 있어서 기존의 방법론에서는 점증적 갱신이 불가능함으로, 학습 데이터에 대해 처음부터 각각의 데이터 개수(21, 30, 60, 90)만큼의 실험을 다시 실시하였으나, 본 논문에서 제안한 방법론은 점증적 갱신이 가능하므로 21개의 학습 데이터로 규칙기반을 생성한 후 분류하고, 이것을 이용하여 학습 데이터 9개, 30개, 30개를 차례로 추가하여 규칙기반을 점증적으로 갱신한 후 분류를 한 결과이다.

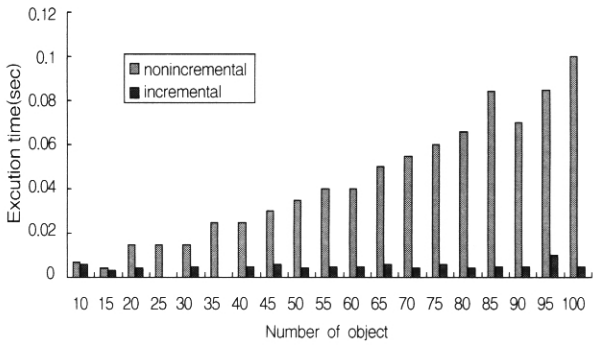
● **점증적 갱신**

빈번한 갱신이 발생할 때 점증적 갱신과 비점증적 갱신에 따른 실행 시간을 비교함으로써 제안된 알고리즘의 효율성을 확인한다. 먼저 10개의 데이터를 이용하여 직사각형 분해를 한 후, 5개씩 새로운 데이터를 추가하여 직사각형을

〈표 5〉 기존의 방법과 제안된 시스템에 의해 생성되는 퍼지 규칙의 수와 분류율 비교

학습 데이터 수	단순 퍼지 격자[12]	분산된 퍼지 격자[12]	CF criterion[12]	NM criterion[12]	RM criterion[12]	Jang[13]	제안된 알고리즘
21	91.3(455)	92.4(8328)	91.1(253)	89.8(71)	89.6(72)	88.3(32)	**90.8(14)**
30	92.7(1727)	93.8(20512)	91.8(307)	93.0(83)	93.3(87)	91.6(36)	**92.8(17)**
60	93.5(2452)	95.4(63069)	93.8(449)	93.9(105)	94.1(107)	93.3(46)	**94.5(24)**
90	94.5(3440)	95.8(140498)	95.1(528)	94.8(150)	94.6(150)	95.0(46)	**95.6(28)**

분해한다. 이때 점증적 갱신과 비점증적 갱신의 실행시간을 비교하면 (그림 7)과 같다. 그래프에서 보는 바와 같이 데이터의 수가 증가할수록 점증적 갱신의 효율성이 크게 증가함을 알 수 있다.



(그림 7) 점증적 갱신과 비점증적 갱신의 비교

5. 결론 및 향후 연구과제

본 논문에서는 동적으로 변화하는 대용량의 데이터베이스로부터 효율적인 데이터 마이닝을 수행하기 위하여 점증적 갱신이 가능한 기존의 직사각형 분해 알고리즘을 보다 효율적으로 개선한 새로운 알고리즘을 제안하였다. 기존의 방법론은 이진 행렬을 일반 그래프로 변환한 후, 변환된 일반 그래프에서 최대 클리크를 찾음으로써 직사각형 분해를 수행하였으나, 일반 그래프에서 최대 클리크를 찾는 문제는 NP-하드 문제이다. 따라서 본 논문에서는 이진 행렬을 그래프로 변환하여 얻은 이분 그래프에서 이분클리크를 찾음으로써 직사각형 분해를 수행하는 개선된 알고리즘을 제안하였다(이분 그래프에서 최대 이분클리크를 찾는 문제는 노드-삭제 방법을 사용하면 다항 시간 안에 풀 수 있다). 제안된 알고리즘은 새롭게 유도된 수학적 정리를 바탕으로 출발하였으므로 그 타당성이 보장되며, 실험을 통하여 점증적 갱신의 효율성을 보이고, 기존의 분류 방법론과의 비교를 통해 제안된 방법론이 규칙의 수와 분류율면에서 우수함을 보였다. 따라서 제안된 알고리즘은 보다 효율적인 점증적 갱신이 가능한 직사각형 분해 알고리즘일 뿐만 아니라, 실세계의 대용량 데이터에 적용이 가능하다. 향후 제안된 알고리즘을 실세계 문제에 적용하기 위한 연구 및 다양한 데이터베이스로의 확장에 대한 연구가 요구된다.

참 고 문 헌

[1] M-S. Chen, J. Han and P. Yu, "Data Mining : An Overview from Database Perspective," IEEE Transactions on Knowledge and Data Engineering, Vol.8, No.6, pp.866-883, 1996.

[2] J. G. Ganascia, Charade, "Apprentissage de bases de connaissances," Induction Symbolique et Numérique à Partir de Données, Cépadues-édition, pp.309-326, 1991.

[3] M. Maddouri, A. Jaoua, "Incremental Learning : Proposition and evaluation of methods," Proceedings of the Joint International Conference on Information Science, JICIS '95, 1995.

[4] V. Novak, "Fuzzy Sets and their Applications," Adam Hilger, Bristol, 1989.

[5] P. P. Wang, "Advances in Fuzzy Theory and Technology," Duke University Edition, 1994.

[6] L. A. Zadeh, "Fuzzy Sets and their Applications to Pattern Classification and Clustering Analysis. Classification and Clustering," Academic Press, New York, 1977.

[7] M. Maddouri, S. Elloumi, Jaoua A, "An Incremental Learning System for Imprecise and Uncertain Knowledge Discovery," Information Sciences, Vol.109, pp.149-164, 1998.

[8] R. Khcherif, A. Jaoua, "Rectangular Decomposition Heuristics for Documentary Databases," Information Sciences, Vol.102, pp.187-202, 1997.

[9] D. S. Hochbaum, "Approximating Clique and Biclique Problems," Journal of Algorithm, Vol.29, pp.174-200, 1998.

[10] M. Yannakakis, "Node deletion problems on bipartite graphs," SIAM J. Comput, Vol.10, pp.310-327, 1981.

[11] D. S. Hochbaum, "Approximating Clique and Biclique Problems," Journal of Algorithms. Vol.29, pp.174-200, 1998.

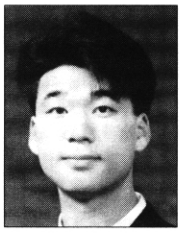
[12] H. Ishibuchi, K. Nozaki, H. Tanaka, "Effective fuzzy partition of pattern space for classification problems," Fuzzy Sets and Systems, Vol.59, pp.295-304, 1993.

[13] Jang, D-S., Choi, H-I., "Automatic Generation of Fuzzy Rules with Fuzzy Associative Memory," Proceeding of the ISCA 5th International Conference, pp.182-186, 1996.

[14] Morton Nadler, Eric P. Smith, "Pattern Recognition Engineering," John Wiley & Sons, pp.341-344, 1997.

[15] M. Maddouri, A. Jaoua, "Comparative empirical evaluation of the discovered knowledge," Proceeding of the IEA/AIE '97 10th International Conference, pp.7-10, 1997.

[16] S. M. Weiss, C. A. Kulikowski, "Computer Systems that Learn," Morgan Kaufmann, Los Altos, CA, 1991.



송 지 영

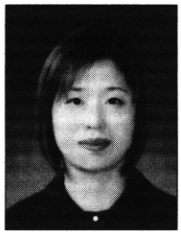
e-mail : songjy@korea.ac.kr

1996년 고려대학교 전산학과(학사)

1999년 고려대학교 전산학과(석사)

1999년~현재 고려대학교 전산학과 박사 과정

관심분야 : 데이터 마이닝, 인공지능, 인공신경망, 생체인식, SVM



임 영 희

e-mail : yheeim@dju.ac.kr
1994년 고려대학교 전산학과(학사)
1996년 고려대학교 전산학과(석사)
2001년 고려대학교 전산학과(박사)
2001년~현재 대전대학교 컴퓨터공학부 강의전담교수
관심분야 : 인공지능, 정보 검색, 텍스트 마이닝, 데이터 마이닝, 데이터베이스 보안

박 대 희

e-mail : dhpark@korea.ac.kr
1982년 고려대학교 수학과(학사)
1984년 고려대학교 수학과(석사)
1989년 플로리다 주립대학 전산학과(석사)
1992년 플로리다 주립대학 전산학과(박사)
1993년~현재 고려대학교 컴퓨터정보학과 교수
관심분야 : 지능 데이터베이스, 데이터 마이닝, 인공신경망, 퍼지 이론