▩ 연구논문

목표 속성을 고려한 연관규칙과 분류 기법
- Directed Association Rules Mining and Classification -

한 경 록*
Han, Kyong Rok
김 재 련**
Kim, Jae Yearn

Abstract

Data mining can be either directed or undirected. One way of thinking about it is that we use undirected data mining to recognize relationship in the data and directed data mining to explain those relationships once they have been found. Several data mining techniques have received considerable research attention. In this paper, we propose an algorithm for discovering association rules as directed data mining and applying them to classification. In the first phase, we find frequent closed itemsets and association rules. After this phase, we construct the decision trees using discovered association rules. The algorithm can be applicable to customer relationship management.

1. 서 론

1.1 연구 배경과 목적

대용량 데이터베이스에서의 지식 발견(knowledge discovery in databases)이라고 정의되는 데이터 마이닝은 감추어져 있긴 하나 잠재적 사용가치가 큰 패턴이나 추세 및 흥미로운 규칙을 발견하는 과정이다. 데이터 마이닝은 directed data mining(DDM)과 undirected data mining(UDM)으로 나눌 수 있다[2,3]. 두 방식의 차이점은 DDM은 특정한 목표 속성(target attribute)을 고려하고, UDM은 목표 속성이나 미리 정의된 클래스를 사용하지 않는다는 것이다. 데이터 마이닝 기법의 관점에서 보면, DDM과 UDM의 차이는 데이터 마이닝 모델을 만들 때 나타난다. DDM은 모델의 결과물(output)이 모델을 만들기 이전에 구체화되고, UDM은 모델 스스로가 결과물을 결정한다. DDM의 기법으로는 사례 기반 추론(memory-based reasoning), 분류(classification), 유전 알고리즘(genetic algorithm) 등이 있고, UDM의 기법은 연관규칙(association rules), 군집화(clustering) 등이 있다. 물론 두 방식 모두에 사용할 수 있는 기법들도 있다. 특히 연관규칙은 두 방식 모두 가능한데[5], 기존의 대부분의 연관규칙 관련 연구들은 UDM에 초점을 맞추고 있다. 또한, 데이터 마이닝 기법을 두 가지 이상 통합하는 연구도 미비한 실정이다.

* 한양대학교 산업공학과 박사과정
** 한양대학교 산업공학과 교수

데이터 마이닝 기법들 중에서 연관규칙(association rules)은 어떤 사건들이 함께 발생하거나, 하나의 사건이 다른 사건을 암시하는 것과 같은 사건간의 상호관계를 나타낸다[14]. 하나의 트랜잭션을 여러 개의 항목들(items)의 집합으로 보고 이러한 트랜잭션들이 하나의 집합으로 주어지면, 연관규칙은 "X⇒Y"라는 형태로 표현되며 여기서 X와 Y는 항목들의 집합이다. 그러한 연관규칙의 의미는 X를 포함하는 트랜잭션들이 Y 또한 포함하는 경향이 있다는 뜻이다.

예를 들어, "라면을 구입하는 고객 중에서 90%의 고객이 김치를 같이 구입한다." 또는 "모든 트랜잭션들 중에서 5%는 맥주와 새우깡을 함께 포함하고 있다." 라는 정보는 연관규칙이다. 여기서 90%를 신뢰도(confidence)라 하고, 5%를 지지도(support)라고 부른다. 연관규칙 발견 문제는 사용자가 지정한 최소지지도와 최소신뢰도의 조건을 만족하는 모든 연관규칙을 찾는 문제이다.

연관규칙을 발견하는 문제는 다음의 두 가지 하위 문제로 세분화된다[4].

1) 사용자가 지정한 최소지지도 이상의 지지도를 갖는 항목집합들(itemsets)을 찾는 단계이다. 항목 집합에 대한 지지도란 그 항목집합을 포함하는 트랜잭션들의 수를 의미한다. 여기서 최소지지도를 만족하는 항목집합을 빈발(large or frequent) 항목집합이라 부르며, 그 이외의 항목집합은 비빈발(small) 항목집합이라고 한다.

2) 빈발 항목집합들을 이용하여 규칙을 생성하는 단계이다. 예를 들어, ABCD와 AB가 빈발 항목집합이면, "신뢰도=지지도(ABCD)/지지도(AB)"의 비율을 계산함으로써 "AB⇒CD"와 같은 연관규칙을 생성시킬 수 있다. 만약 "신뢰도≥최소신뢰도"이면 그 규칙은 강한(strong) 연관규칙이라고 부르며 사용자에게는 잠재적 사용가치가 큰 정보로 인식된다.(이 규칙은 ABCD가 빈발이기 때문에 최소지지도를 가질 것이다.)

본 논문에서는 DDM으로서의 연관규칙 발견에 초점을 두고, 발견된 연관규칙을 분류 기법에 적용한다. 제안하는 알고리즘은 크게 두 단계로 나누어진다. 첫 단계에서는 목표 속성을 고려하면서 빈발 closed 항목집합을 이용하여 발견되는 빈발 항목집합과 연관규칙의 수를 줄인다. 두 번째 단계에서는 찾아진 빈발 항목집합과 연관규칙을 이용하여 먼저 분지할 속성을 순차적으로 선택하고, 새로운 트랜잭션이 발생할 때 정해진 분류 기준에 의하여 분석을 할 수 있도록 한다.

1.2 기존 연구

연관규칙 문제의 대표적인 알고리즘이 Apriori 알고리즘이다[4]. 이 알고리즘에서 사용하는 개념을 바탕으로 항목간의 연관관계를 찾는 연구가 진행되었고, 항목의 수량을 고려하거나[15] 항목에 제약을 주는[16] 알고리즘들이 개발되었다. 또 다계층간의 항목들에 대한 연관규칙을 발견하거나[7] 항목간의 연관성에 시간이라는 속성을 첨가하여 주기적인 패턴을 갖는 연관규칙을 찾아내기도 했다[6,13]. 최근에는 인터넷 관련 기술이 빠르게 발전함에 따라 웹에서 사용자의 패턴을 발견하고자 하는 연구도 활발하고, 데이터 마이닝과 상호보완적인 관계가 있는 OLAP(On-Line Analytical Processing)와 마이닝 기법들과의 통합도 다양하게 논의되고 있다[8]. 또한 Galois connection에 기초한 closure 메커니즘을 이용하여 빈발 closed 항목집합을 발견함으로써 빈발 항목집합과 연관규칙의 수를 줄이는 알고리즘이 연구되었다[9,11,12]. 속성들의 정보량을 계산하여 먼저 분지할 속성을 정하는 ID3 분류 방법[1]이 개발되었다.

본 연구는 다음과 같이 구성되어 있다. 2장은 연관규칙과 분류에 대한 개념을 설명하고, 3장에서는 본 연구가 제안하는 알고리즘을 논의한다. 4장은 제안하는 알고리즘을 적용하여 예제를 소개한다. 5장은 본 연구의 결론을 기술한다.

2. 연관규칙과 분류

2.1 연관규칙

$I=\{i_1,i_2,\cdots\cdots,i_m\}$를 항목(item)이라 불리는 문자들의 집합이라고 하자. D는 트랜잭션들의 집합이고 각 트랜잭션 T는 $T\subseteq I$인 항목들의 집합이라고 하자. 한 트랜잭션에서 구입하는 항목들의 수량은 고려하지 않기로 가정한다. I의 원소인 항목들의 집합(itemset)을 X라 할 때 "$X\subseteq T$이면 트랜잭션 T가 X를 포함한다"라고 말한다. 연관규칙은 "$X\Rightarrow Y$"의 형태로 표시되고, 여기서 $X\subset I$, $Y\subset I$ 및 $X\cap Y=\varnothing$이다[4,5,10].

X를 포함하는 D에 있는 트랜잭션들의 c%가 Y 또한 포함하고 있으면 연관규칙 $X\Rightarrow Y$는 트랜잭션들의 집합 D에서 신뢰도 c를 가지고 있다는 뜻이다. D에 있는 트랜잭션들의 s%가 $X\cup Y$를 포함하면 연관규칙 $X\Rightarrow Y$는 트랜잭션들의 집합 D에서 지지도 s를 가지고 있음을 의미한다. 최소지지도 이상을 갖는 항목집합을 빈발 항목집합이라 한다. k개의 항목들로 이루어진 빈발 항목집합을 빈발 k-항목집합이라 한다. 빈발 k-항목집합들의 집합을 L_k라 하고, 이를 생성하기 위한 후보 k-항목집합들의 집합을 C_k(잠재적 빈발 항목집합)라 한다. D가 주어졌을 때, 연관규칙 문제는 최소지지도와 최소신뢰도 이상의 지지도와 신뢰도를 갖는 모든 빈발 항목집합들과 연관규칙을 생성하는 문제이다.

2.2 Closed Itemsets

연관규칙을 발견하는 데는 두 가지 문제점이 있다. 첫째는 많은 빈발 항목집합을 생성시키게 되는데, 특히 최소지지도가 낮아질수록 더욱 그렇다. 둘째는 그에 따른 연관규칙 또한 많다는 것이다. 이 문제에 대한 대안으로 빈발 closed 항목집합을 발견하는 알고리즘이 있다[12]. 예를 들어, 데이터베이스에 $\{(a_1, a_2, \ldots, a_{100}), (a_1, a_2, \ldots, a_{50})\}$이라는 두 개의 트랜잭션이 있다고 하자. 최소지지도는 1이고, 최소신뢰도는 50%라고 하자. 기존의 알고리즘으로 구하면 (a_1), ..., (a_{100}), (a_1, a_2), ..., (a_{99}, a_{100}), ..., $(a_1, a_2, \ldots, a_{100})$과 같이 $2^{100}-1$(약 10^{30})개의 빈발 항목집합이 발견된다[9].

그러나 closed 개념을 사용하면 $\{(a_1, a_2, \ldots, a_{50}), (a_1, a_2, \ldots, a_{100})\}$과 같이 두 개의 빈발 closed 항목집합만이 생기고, 거기에 따른 연관규칙도 "$(a_1, a_2, \ldots, a_{50})\Rightarrow(a_{51}, a_{52}, \ldots, a_{100})$"으로서 하나만 만들어진다. 다른 모든 빈발 항목집합에 대한 연관규칙은 위의 빈발 closed 항목집합과 그에 따른 연관규칙으로부터 유도할 수 있다. 따라서 모든 빈발 항목집합을 발견하는 대신에 정보의 손실 없이 빈발 closed 항목집합만을 발견한다.

<표 1>에 트랜잭션 데이터베이스가 주어져 있다. 최소지지도를 2라고 가정한다. 즉, 5개의 트랜잭션들 중에서 2번 이상 발생하면 빈발이다. TID는 Transaction IDentifier이다. <표 1>에서 1번 고객이 a, c, d, e, f의 5개의 항목을 구입했다는 것을 알 수 있다. 아래의 데이터베이스에 Apriori 알고리즘을 적용하면 빈발 1-항목집합에서 빈발 4-항목집합까지 총 20개의 빈발 항목집합들을 찾을 수 있다.

<표 1> 트랜잭션 데이터베이스

TID	Items
1	a c d e f
2	a b e
3	c e f
4	a c d f
5	c e f

그러나 빈발 closed 항목집합은 {acdf, cef, ae, cf, a, e}로서 6개만 발견된다. 예를 들어, 항목을 기준으로 보면 a, c, d, f의 4개의 항목을 모두 구입한 고객은 1번과 4번이고, 반대로 트랜잭션 기준으로 1번과 4번 고객은 공통으로 구입한 항목이 a, c, d, f 이외에는 없다. 이처럼 어떤 항목집합에 대해 그 항목집합을 구입한 고객을 모두 찾아낸 후에 다시 그 고객들이 그 항목집합 이외에는 공통으로 구입한 항목이 없으면 그 항목집합을 closed 항목집합이라고 한다. 다시 말해서 a와 e를 같이 구입한 고객은 1번과 2번 고객이고, 두 고객은 a와 e 항목 이외에는 공통인 항목인 없으므로 {a, e}는 closed 항목집합인 것이다. 물론 {a, e}는 지지도가 2이므로 최소지지도를 만족하여 빈발 closed 항목집합이라고 한다. 하지만 d 항목은 구입한 고객이 1번과 4번이지만 두 고객은 d 항목이외에도 a, c, f 항목이 공통이어서 {d}를 closed 항목집합이라고 할 수 없는 것이다.

2.3 분류

분류 기법은 새로 발생한 트랜잭션의 속성들을 조사하여 그 트랜잭션을 미리 정의한 클래스들의 집합 중 하나로 할당하는 문제이다. 상위의 뿌리 노드로부터 하위의 말단 노드까지 따라 내려가는 의사 결정 나무의 형태가 많이 사용되며, 간결하고 효율적인 나무를 만들기 위해 어떤 속성으로 먼저 분지할 것인가 하는 문제가 발생한다. 먼저 분지할 속성들의 순서 결정 방식에 따라 의사 결정 나무의 크기가 달라진다. ID3 알고리즘은 의사 결정 나무를 만들기 위해 정보량s 을 계산한다. 각각의 노드에서 사용할 수 있는 속성들을 대상으로 gain값을 계산하여 가장 값이 큰 속성으로 먼저 분지를 하고, 다음 수준(level)에서는 그 속성을 제외한 나머지 속성들을 이용하여 정보량을 계산하는데, 이 과정을 모든 예제 트랜잭션이 말단 노드에 속할 때까지 반복한다.

3. 제안하는 알고리즘

3.1 기호 및 용어 설명

D : 트랜잭션들의 집합
N : 항목들의 수
T : D의 트랜잭션들의 수
I : 항목들의 집합
Y : 목표 속성
X : 목표 속성을 제외한 나머지 속성들
F : 첫 번째 시행에서 빈발하는 항목들의 집합
k-부분집합 : k개의 원소가 있는 부분집합
k-항목집합 : k개의 항목들을 갖는 항목집합
L_k^c : 빈발 closed k-항목집합들의 집합
Minimum support(Minsup.) : 최소지지도
Minimum confidence(Minconf.) : 최소신뢰도
Tree-1 : 특정 목표 속성값을 뿌리 노드로 갖는 트리
Tree-2 : 전체 트랜잭션을 분류한 트리
$X \Rightarrow Y(s_1,s_2,c)$: s_1(X의 지지도: 개수), s_2(X∪Y의 지지도: 개수), c(신뢰도: 백분율)

3.2 알고리즘 설명

본 논문에서는 DDM으로서의 연관규칙 발견에 초점을 두고, 발견된 연관규칙을 분류 기법에 사용한다. 제안하는 알고리즘은 크게 두 단계로 나누어진다. 첫 단계에서는 목표 속성을 고려하면서 사용자가 정의한 최소지지도를 만족하는 빈발 closed 항목집합과 최소신뢰도를 만족하는 연관규칙을 발견한다. 두 번째 단계에서는, 찾아진 빈발 항목집합과 연관규칙을 이용하여 먼저 분지할 속성을 순차적으로 선택하여 트리를 만들고, 새로운 트랜잭션이 발생할 때 정해진 분류 기준에 의하여 분석을 한다.

다음은 알고리즘의 전개를 보여준다.

단계 1. 트랜잭션들의 집합 D에 있는 속성들 중에서 하나의 목표 속성(Y)을 선택한다. 즉 연관규칙을 "X⇒Y"라 할 때, Y 부분을 정하는 것이다. 또한 최소지지도와 최소신뢰도를 정한다.

단계 2. 각 속성이 두 가지 이상의 값을 가지므로, 범주형(categorical) 데이터와 수량형(quantitative) 데이터를 같이 고려한다. 전자는 각 범주에 대해 직접 정수 변환(mapping)을 하고 후자는 연속적(continuous) 데이터이므로 일단 구간으로 나눈 후에 각 구간을 정수 변환하는 방법을 택한다.

단계 3. 트랜잭션들의 집합 D에 대해 각 항목들이 몇 번씩 발생하는지 지지도를 계산한다. 그리고 최소지지도(Minsup.)를 만족하는 항목만을 선택하여 빈발 1-항목집합(F)이라 하고, 지지도가 높은 순서대로 정렬한다.

단계 4. 단계 3에서 정렬한 항목들의 순서를 고려하여, D의 각 트랜잭션들에 있는 목표 속성을 제외한 나머지 속성값들을 지지도가 높은 순서대로 재배치한다.

단계 5. 빈발 1-항목집합들 중에서 가장 낮은 지지도를 갖는 항목부터 conditional 데이터베이스를 구성한 뒤에 빈발 closed 항목집합(L_k^c)을 발견한다. closed 개념은 후보 항목집합을 만들 필요가 없고, 발견된 빈발 항목집합과 그에 따른 연관규칙의 수가 줄어들면서도 기존의 연관규칙 발견 알고리즘이 갖는 모든 정보를 그대로 유지한다.

단계 6. 발견된 빈발 closed 항목집합을 이용하여 연관규칙을 생성한다. 이렇게 만들어진 연관규칙(IF... THEN...)은 THEN 부분에 오로지 하나의 목표 속성만을 갖는다. 연관규칙이 최소신뢰도(Minconf.)를 만족하면 강한 연관규칙이라고 한다.

단계 7. 발견된 연관규칙을 이용하여 목표 속성의 속성값을 뿌리 노드(0-level)로 하여 1-level에서는 빈발 1-항목집합을 고려하고 2-level에서는 빈발 2-항목집합을 고려하는 방식으로 Tree-1을 형성해 나간다. 각 level에서는 신뢰도가 높은 항목부터 분지한다.

단계 8. 마지막으로 전체 트랜잭션을 분류하는 Tree-2를 만든다. 빈발 1-항목집합 중에서 신뢰도가 높은 것부터 뿌리 노드로 선택하고 신뢰도가 같으면 지지도를 비교하여 높은 것을 택한다. 각 말단 노드가 같은 속성값을 가질 때까지 분지해 나간다. Tree-2를 이용하여 새로운 트랜잭션에 대한 분류를 실시한다.

ID3는 모든 트랜잭션을 고려하기 때문에 희박한 빈도수를 갖는 트랜잭션도 포함하지만, 제안하는 분류 방식은 일정 지지도 이상을 만족하는 트랜잭션만이 의미가 있다. 첫 단계에서 발견하는 빈발 closed 항목집합과 연관규칙은 DB의 각 속성들 사이의 연관성을 파악할 수 있고, 다시 그것들을 고객 분류에 이용을 하는 것이다. 연관규칙을 바탕으로 한 분류는 특정한 목표 속성값에 대한 분류 정보를 빨리 얻을 수 있으며, 희박한 발생 빈도수를 갖는 트랜잭션을 고려하지 않으므로 트리의 크기가 작아진다.

4. 예 제

다음의 <표 2>는 고객 신용을 평가한 데이터베이스이다[1]. 이 예제에서는 목표 속성으로 "RISK"를 선택하고, 3번 이상 발생하면 빈발로 가정한다. 즉 최소지지도는 3이다. <표 3>에서는 각 속성을 문자나 정수를 사용하여 간단한 표현으로 변환한다. 예를 들어 "A1"은 "RISK=high"를 의미하고, "E2"는 "INCOME=$15 to $35k"를 나타낸다.

<표 2> Data from Credit History of Loan Application

NO.	RISK	CREDIT HISTORY	DEBT	COLLATERAL	INCOME
1	high	bad	high	none	$0 to $15k
2	high	unknown	high	none	$15 to $35k
3	moderate	unknown	low	none	$15 to $35k
4	high	unknown	low	none	$0 to $15k
5	low	unknown	low	none	over $35k
6	low	unknown	low	adequate	over $35k
7	high	bad	low	none	$0 to $15k
8	moderate	bad	low	adequate	over $35k
9	low	good	low	none	over $35k
10	low	good	high	adequate	over $35k
11	high	good	high	none	$0 to $15k
12	moderate	good	high	none	$15 to $35k
13	low	good	high	none	over $35k
14	high	bad	high	none	$15 to $35k

<표 3> 속성 변환 과정

RISK:A	high(1): A1	moderate(2): A2	low(3): A3
CREDIT HISTORY:B	bad(1): B1	unknown(2): B2	good(3): B3
DEBT:C	high(1): C1	low(2): C2	
COLLATERAL:D	none(1): D1	adequate(2): D2	
INCOME:E	$0 to $15k(1): E1	$15 to $35k(2): E2	over $35k(3): E3

각 속성의 지지도를 계산하여 최소지지도를 만족하는 속성을 찾은 후에 높은 순서로 정렬하면, D1(11), C1(7), C2(7), E3(6), B2(5), B3(5), B1(4), E1(4), E2(4), D2(3)이다. <표 2>의 목표 속성을 제외한 나머지 속성들을 지지도가 높은 순서대로 나타낸 것이 <표 4>이다. 즉, 1번 트랜잭션은 D1, C1, B1, E1의 순서로 재배치된다.

<표 4> 속성 재배치

NO	속성 재배치				목표 속성
1	D1	C1	B1	E1	A1
2	D1	C1	B2	E2	A1
3	D1	C2	B2	E2	A2
4	D1	C2	B2	E1	A1
5	D1	C2	E3	B2	A3
6	C2	E3	B2	D2	A3
7	D1	C2	B1	E1	A1
8	C2	E3	B1	D2	A2
9	D1	C2	E3	B3	A3
10	C1	E3	B3	D2	A3
11	D1	C1	B3	E1	A1
12	D1	C1	B3	E2	A2
13	D1	C1	E3	B3	A3
14	D1	C1	B1	E2	A1

<표 5> conditional database

D2-cond.(3)

C2	E3	B2	A1
C2	E3	B1	A2
C1	E3	B3	A2

E2-cond.(4)

D1	C1	B2	A1
D1	C2	B2	A2
D1	C1	B3	A2
D1	C1	B1	A1

E1-cond.(4)

D1	C1	B1	A1
D1	C2	B2	A1
D1	C2	B1	A1
D1	C1	B3	A1

B1-cond.(4)

D1	C1	A1
D1	C2	A1
C2	E3	A2
D1	C1	A1

B3-cond.(5)

D1	C2	E3	A3
C1	E3		A3
D1	C1		A1
D1	C1		A2
D1	C1	E3	A3

B2-cond.(5)

D1	C1		A1
D1	C2		A2
D1	C2		A1
C2	E3		A3
D1	C2	E3	A3

E3-cond.(6)

D1	C2	A3
C2		A3
C2		A2
D1	C2	A3
C1		A3
D1	C1	A3

C2-cond.(7)

D1	A2
D1	A1
D1	A3
	A3
D1	A1
	A2
D1	A3

C1-cond.(7)

D1	A1
D1	A1
	A3
D1	A1
D1	A2
D1	A3
D1	A1

D1-cond.(11)

A1	A2	A3
A1	A2	A3
A1		A3
A1		
A1		
A1		

<표 5>는 지지도가 가장 낮은 속성부터 conditional 데이터베이스를 만들어서 빈발 closed 항목집합을 찾는다. 발견된 빈발 closed 항목집합은 <그림 1>과 같다.

D1E1A1(4) D1B1A1(3) E3B3A3(3) D1E3A3(3) C2E3A3(3) D1C1A1(4) D1A1(6)

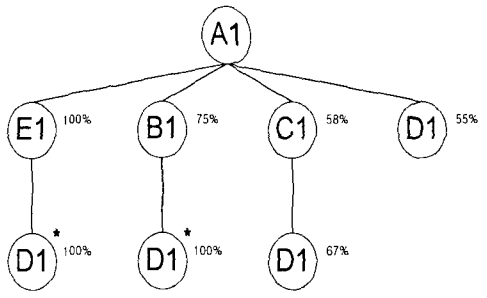
<그림 1> 빈발 closed 항목집합

다른 모든 빈발 항목집합과 그에 따른 연관규칙은 위의 빈발 closed 항목집합을 이용하여 유도할 수 있다. <그림 1>을 이용하여 <그림 2>의 연관규칙들을 생성한다. "C2E3⇒A3(4,3,75%)"의 의미는 C2E3라는 속성을 갖는 고객은 A3라는 클래스로 구분되며 C2E3를 포함하는 트랜잭션의 수가 4이고 C2E3A3를 포함하는 트랜잭션의 수가 3으로서 신뢰도는 75%(3/4×100)라는 뜻이다. 만약 최소신뢰도를 50%라고 한다면 이 규칙은 강한 연관규칙이라고 한다.

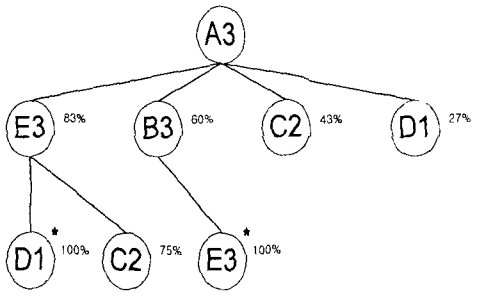
D1⇒A1(11,6,55%)	D1⇒A3(11,3,27%)
C1⇒A1(7,4,58%)	C2⇒A3(7,3,43%)
B1⇒A1(4,3,75%)	B3⇒A3(5,3,60%)
E1⇒A1(4,4,100%)	E3⇒A3(6,5,83%)
D1C1⇒A1(6,4,67%)	C2E3⇒A3(4,3,75%)
D1B1⇒A1(3,3,100%)	D1E3⇒A3(3,3,100%)
D1E1⇒A1(4,4,100%)	E3B3⇒A3(3,3,100%)

<그림 2> 연관규칙

연관규칙을 이용하여 목표 속성값을 뿌리 노드로 하는 트리가 <그림 3>과 <그림 4>에 있다. 이 예제에서는 A1과 A3을 뿌리 노드로 한다. 최소신뢰도를 고려하여 강한 연관규칙만으로 트리를 만들 수도 있지만 여기서는 모든 연관규칙을 트리로 표현한다. 트리의 뿌리 노드를 0-level로 하고 빈발 1-항목집합으로 1-level을 구성하며 빈발 2-항목집합으로 2-level을 만든다. 즉, 빈발 n-항목집합은 깊이가 n인 트리를 형성한다.



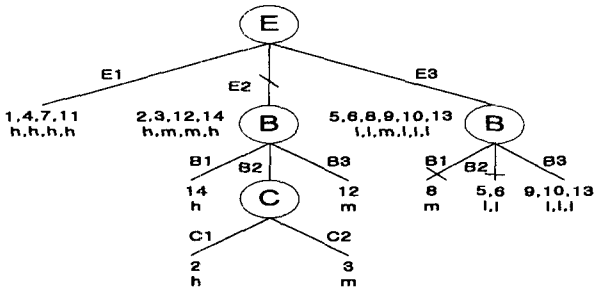
<그림 3> A1을 고려한 Tree-1



<그림 4> A3을 고려한 Tree-1

각 레벨에서는 신뢰도가 높은 속성부터 분지한다. 분지할 속성들의 신뢰도가 같으면 지지도를 살펴보고, 지지도까지 같으면 먼저 만들어진 빈발 항목집합을 우선적으로 분지한다. <그림 3>에서 새로운 트랜잭션에 대해 이 고객이 A1(RISK:high)인지를 알아보려면 깊이 우선 방법으로 트리를 탐색한다. "*" 표시는 신뢰도 100%를 의미한다.

<그림 5>는 전체 트랜잭션을 분류한 트리를 보여 주는데, 어떤 속성으로 먼저 분지할 것인가는 빈발 1-항목집합에 의해 생성된 <그림 2>의 연관규칙들에 대한 신뢰도를 고려하여 가장 높은 신뢰도를 갖는 속성부터 분지한다. 이 예제에서는 항목 E, B, C의 순서로 분지한다.



<그림 5> 전체 트랜잭션 분류 트리(Tree-2)

이렇게 만들어진 트리는 최소지지도를 이미 고려했으므로 최소한 3번 이상 발생해야 관심을 갖게 되고 결국 말단 노드가 3개 미만의 목표 속성값으로 이루어진다면 절단(pruning)한다. 따라서 <그림 5>에서 E2 속성의 하위 노드와 E3 속성으로 분지한 다음의 B1, B2 속성은 최소지지도를 만족하지 못하므로 절단한다.

5. 결 론

본 논문에서는 directed 데이터 마이닝으로서의 연관규칙에 초점을 두고, 발견된 연관규칙을 분류 기법에 적용함으로써 두 가지 마이닝 기법의 통합을 제안한다. 제안하는 알고리즘은 첫 단계에서 목표 속성을 고려하면서 closed 개념을 이용하여 발견되는 빈발 항목집합과 연관규칙의 수를 줄인다. 두 번째 단계에서는 찾아진 빈발 항목집합과 연관규칙을 이용하여 분지할 속성의 순서를 정하고 트리를 만든다. 연관규칙을 기반으로 분류된 데이터들은 신용카드 사기 고객 관리나 환율변동 예측, 금융권의 신용과 외환 위험관리는 물론 의학 분야에서 암, 심장마비 등의 질병을 자동 진단하는 데에도 이용할 수 있다. 특히 인터넷 뱅킹이나 인터넷 쇼핑몰이 발전함에 따라 고객 데이터베이스를 얼마나 잘 관리하느냐에 관심이 높아지고 있다. 대용량 데이터베이스에서 발견된 연관규칙을 이용하여 고객이탈을 방지하고 잠재고객을 발굴하는 것은 물론 고객을 분류하는데도 유용하다.

그리고 본 연구에서는 발견된 빈발 항목집합과 연관규칙을 이용하여 트리를 만드는 방법을 제안하였으나 앞으로는 대용량의 데이터를 사용하여 실험을 해서 지지도 변화에 따른 트리 구성 시간이나 구성된 트리가 얼마나 새로운 트랜잭션을 잘 분류해 내는지에 관한 트리의 에러율에 대해서도 연구되어야 할 것이다.

참고문헌

[1] George F. Luger, William A. Stubblefield, ARTIFICIAL INTELLIGENCE, 2nd Edition, Addison-Wesley, U.S.A., 1993.
[2] Michael J. A. Berry, Gordon Linoff, Data Mining Techniques, WILEY, U.S.A., 1997.

[3] Pieter Adriaans and Dolf Zantinge, DATA MINING, Addison-Wesley, Harlow, U.K., 1996.
[4] Agrawal, R., Srikant, R., Fast Algorithms for Mining Association Rules, In Proc. of the 20th Int'l Conference on Very Large Databases, Santiago, Chile, September 1994.
[5] Bing Liu, Wynne Hsu, Yiming Ma, Integrating Classification and Association Rule Mining, Proc. of 4th Int. Conf. on Knowledge Discovery and Data Mining (KDD-98), New York, USA, 1998.
[6] Han, J., Gong, W., Yin, Y., Mining segment-wise periodic patterns in time-related databases, In Proc. 1998 Int. Conf. on Knowledge Discovery and Data Mining(KDD'98), New York City, NY, August 1998.
[7] J. Han and Y. Fu, Discovery of Multiple-Level Association Rules from Large Databases, Proc. of 1995 Int'l Conf. on Very Large Data Bases (VLDB'95), Zurich, Switzerland, pp. 420-431, Sep. 1995.
[8] J. Han, S. Chee, and J. Y. Chiang, Issues for On-Line Analytical Mining of Data Warehouses, Proc. of 1998 SIGMOD'96 Workshop on Research Issues on Data Mining and Knowledge Discovery(DMKD'98), Seattle, Washington, June, pp. 2:1-2:5, 1998.
[9] J. Pei, J. Han, and R. Mao, CLOSET: An Efficient Algorithm for Mining Frequent Closed Itemsets, Proc. 2000 ACM-SIGMOD Int. Workshop on Data Mining and Knowledge Discovery(DMKD'00), Dallas, TX, May 2000.
[10] M. S. Chen, J. Han, and P. S. Yu, Data Mining: An Overview from a Database Perspective, IEEE Transactions on Knowledge and Data Engineering, 8(6): 866-883, 1996.
[11] M. J. Zaki and C. Hsiao. Charm: An efficient algorithm for closed association rule mining. In Technical Report 99-10, Computer Science, Rensselaer Polytechnic Institute, 1999.
[12] N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal. Discovering Frequent Closed Itemsets for Association Rules, In Proc. 7th Int. Conf. Database Theory(ICDT'99), p.398-416, Jerusalem, Israel, January 1999.
[13] Ozden, B., Ramaswamy, S., Silberschatz, A., Cyclic Association Rules, In Proc. 1998 Int. Conf. Data Engineering(ICDE'98), pp. 412-421, Orlando, FL, Feb. 1998.
[14] R. Agrawal, T. Imielinski, and A. Swami, Mining Association Rules between Sets of Items in Large Databases, Proc. of the ACM SIGMOD Int'l Conference on Management of Data, Washington D.C., pp. 207-216, May 1993.
[15] R. Srikant and R. Agrawal, Mining Quantitative Association Rules in Large Relational Tables, Proc. of the ACM SIGMOD Conference on Management of Data, Montreal, Canada, June 1996.
[16] Srikant, R., Vu, Q., Agrawal, R., Mining Association Rules with Item Constraints, Proc. of the 3rd Int'l Conference on Knowledge Discovery in Databases and Data Mining, Newport Beach, California, August 1997.