▩ 연구논문

데이터 큐브를 이용한 연관규칙 발견 알고리즘

- An Algorithm for Cube-based Mining Association Rules and Application to Database Marketing -

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

Abstract

The problem of discovering association rules is an emerging research area, whose goal is to extract significant patterns or interesting rules from large databases and several algorithms for mining association rules have been applied to item-oriented sales transaction databases. Data warehouses and OLAP engines are expected to be widely available. OLAP and data mining are complementary; both are important parts of exploiting data. Our study shows that data cube is an efficient structure for mining association rules. OLAP databases are expected to be a major platform for data mining in the future. In this paper, we present an efficient and effective algorithm for mining association rules using data cube. The algorithm can be applicable to enhance the power of competitiveness of business organizations by providing rapid decision support and efficient database marketing through customer segmentation.

1. 서 론

1.1 연구 배경

대용량 데이터베이스에서의 지식 발견(knowledge discovery in databases)이라고 정의되는 데이터 마이닝은 감추어져 있긴 하나 잠재적 사용가치가 큰 패턴이나 추세 및 흥미로운 규칙들을 발견하는 과정이다. 데이터 마이닝(Data Mining)에 대한 연구가 활발히 진행되면서, 표면적으로는 관련되지 않아 보이는 데이터들이 새롭고 유용한 정보를 창출하게 되었다[10].

데이터 마이닝은 최근 들어 시장전략 수립, 수요예측, 의료진단, 상품진열 등 광범위한 분야에 응용되고 있으며, 감(feeling)을 사실(fact)로 전환시킬 수 있는 능력이 큰 장점으로 인식되고 있다. 이러한 데이터 마이닝의 기법에는 신경망(neural networks), 분류(classification), 연관규칙(association rules), 순차패턴(sequential pattern), 유전 알고리즘(genetic algorithms), 군집화(clustering)등이 있다[3,11].

* 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. 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.)

데이터 마이닝과 상호보완관계를 형성하며 발전한 OLAP(On-Line Analytical Processing)는 데이터 큐브(Data Cube)를 이용하여 정보를 다양한 각도에서 분석하고 확인하는 역할을 수행한다. "다차원 정보분석"이라는 의미로 정의할 수 있는 OLAP는 OLTP(On-Line Transaction Processing)에 상대되는 개념으로, 오늘날 데이터 웨어하우스(Data Warehouse) 환경에서 데이터 접근 전략의 중요한 요소로 자리잡아가고 있다. 사용자는 정보를 분석하기 위해 비교하기를 원하며 이러한 비교는 다양한 각도에서 수행되어야 한다. OLAP 환경에서 사용자는 정형화된 보고서를 단순히 조회하는 방식에서 벗어나 대화식으로 정보를 분석한다[1,17].

예를 들어, 지역, 영화장르, 성별이라는 3개의 차원으로 구성된 큐브가 있을 때, "서울지역에서 여자들이 관람한 액션영화는 몇 개인가?"와 같은 질문이나 연령, 직업, 취미, 자동차이름이라는 4개의 차원으로 이루어진 데이터 큐브가 주어졌을 때, "취미가 등산인 30대 공무원이 구입한 한양자동차는 모두 몇 대인가?"와 같은 질문은 전형적인 OLAP라고 할 수 있다.

OLAP는 최종 사용자가 다차원 정보에 직접 접근하여 대화식으로 정보를 분석하고 의사결정에 활용하는 과정으로서, 질의에 신속한 응답 성능을 제공한다는 장점이 있어서 데이터 마이닝과 함께 기업의 의사결정을 체계적으로 지원하는 정보 기반인 데이터 웨어하우스 환경과 불가분의 관계를 유지하면서 발전하고 있다.

1.2 연구 목적

지금까지 연관규칙 발견은 요약되지 않은 sales transaction databases를 대상으로 연구가 진행되어져 왔으며 주로 빈발 항목집합들을 찾는 단계에 초점을 맞추고 있다. 따라서 빈발 항목집합을 찾기 위하여 대부분의 연관규칙 발견 알고리즘에서 사용하는 join과 prune 단계에 많은 시간을 투자한다. 또한 항목간의(item-oriented) 연관규칙을 고려했기 때문에 고객의 속성 사이의 연관규칙 발견과 같이 고객의 정보(나이, 성별, 취미 등)를 포함하여 다차원으로 마이닝하는 경우는 드물었다.

DSS(Decision Support Systems)용 질의를 온라인으로 처리하기 위해 데이터 웨어하우스 환경이 구축되기 시작했고, 사용자는 다양한 애플리케이션과 툴(tool)을 통해 데이터 웨어하우스에 저장된 데이터를 조회하고 분석한다. 요즘 사용되는 대표적인 툴로서 OLAP 툴과 데이터 마이닝 툴이 주목받고 있다. 특히 OLAP를 위하여 데이터 표현에 적합한 데이터 큐브를 사용한다[1,2].

본 논문에서는 이미 만들어진 데이터 큐브를 OLAP뿐만 아니라 연관규칙 발견에도 이용하는 알고리즘을 제시한다. 다시 말해서, 지금까지는 데이터 큐브가 OLAP에만 사용되어 왔지만 본 논문은 항목뿐만 아니라 고객의 속성도 차원으로 갖는 데이터 큐브를 이용하여 최소지지도를 만족하는 서브큐브(Subcube)를 발견한 다음에, 실제적으로 의사결정에 필요한 연관규칙을 마이닝하여 고객 세분화를 통한 데이터베이스(DB) 마케팅에의 활용에 목적을 둔다.

1.3 기존 연구

연관규칙 문제의 대표적인 알고리즘이 Apriori 알고리즘이다[4]. 이 알고리즘에서 사용하는 개념을 바탕으로 항목간의 연관관계를 찾는 연구가 진행되었고 항목의 수량을 고려하거나 항목에 제약을 주는 알고리즘들이 개발되었다[15,16]. 또 항목간의 연관성에 시간이라는 속성을 첨가하여 주기적인 패턴을 갖는 연관규칙을 찾아내기도 했다[6,12].

데이터 마이닝 툴이 데이터 사이의 패턴을 찾는 데 중점을 둔 반면, OLAP툴은 데이터를 다각도로 분석하여 빠르게 질의에 응답하는 툴이다[2]. 최근에는 OLAP과 데이터 마이닝을 하나의 시스템으로 통합하려는 시도가 이루어지고 있다[9]. 또한 기존의 마케팅은 불특정 다수의 고객층(또는 구매 예상집단)을 대상으로 하는 매스(Mass) 마케팅이었지만 요즘에는 소비자 요구의 다양화와 생활 양식의 변화로 집단이 아닌 개인으로서의 고객대응(일대일 마케팅)에 대한 연구가 필요하게 되었다[1].

본 연구는 다음과 같이 구성되어 있다. 2장은 연관규칙과 OLAP에 대한 개념을 설명하고, 3장에서는 본 연구가 제안하는 데이터 큐브를 이용하여 연관규칙을 발견하는 알고리즘을 논의한다. 4장은 제안하는 알고리즘을 적용하여 여러 가지 관점에서 예제를 다루어 본다. 5장은 본 연구의 결론을 기술한다.

2. 연관규칙과 OLAP

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 포함한다라고 말한다. 연관규칙은 "$X \Rightarrow Y$"의 형태로 표시되고, 여기서 $X \subset I$, $Y \subset I$ 및 $X \cap Y = \varnothing$이다[5,13].

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[4,8,14]. 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 사용자가 지정한 최소지지도(Minsup.)와 최소신뢰도(Minconf.) 이상의 지지도와 신뢰도를 갖는 모든 빈발 항목집합들과 연관규칙을 생성하는 문제이다.

2.2 Apriori 알고리즘

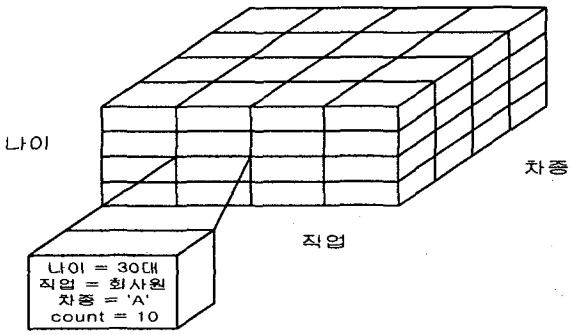
알고리즘의 첫 번째 시행에서는 빈발 1-항목집합들을 결정하기 위해 단순히 모든 트랜잭션을 읽어서 각 항목별로 빈도 수를 계산한다. $k(k \geq 2)$번째 시행부터는 두 가지 단계를 고려한다. 먼저, (k-1)번째 시행에서 발견된 빈발 항목집합들인 L_{k-1}이 후보 항목집합들인 C_k를 발생시키기 위해 사용된다. 다음으로, 데이터베이스가 검색되어 C_k에 있는 후보들의 지지도가 계산되고 최소지지도를 만족하는 C_k만이 L_k로 진입한다.

이러한 시행(pass)이 계속 반복되어 $L_k(k \geq 1)$가 공집합이 되면 알고리즘을 종료한다. 알고리즘에서 가장 중요한 부분은 join과 prune 두 가지 단계로 구성된다. join 단계에서는, C_k를 생성하기 위해 $L_{k-1}*L_{k-1}$(self-join)을 사용하는데, 여기서 *는 연결(concatenation) 연산이다. prune 단계에서는, join 단계에서 생성된 C_k에 있는 모든 항목집합들 $c(c \in C_k)$에 대해 c의 어떤 (k-1)-부분집합이 L_{k-1}에 존재하지 않으면 그 후보 c를 삭제한다[4].

2.3 OLAP

OLAP라는 용어는 1993년 E. F. Codd에 의해 처음 사용된 용어로 OLTP와 상대되는 개념이다. OLTP 시스템은 원시 데이터가 기록되는 시스템으로 '무엇(What)'에 초점을 맞추면서 일상적인 기업의 운영을 지원하는 반면, OLAP 시스템은 이렇게 수집된 데이터를 의사결정에 활용하는 측면을 담당하며 '왜(Why)'에 초점이 맞추어진다. 최근 OLAP는 데이터 웨어하우징(Data Warehousing)이라는 개념과 결합하면서 다차원 정보분석의 필요성에 대한 인식을 확산시키게 되었고 웨어하우스 데이터를 분석하는 가장 기본적인 수단으로 자리잡고 있다[1,17].

다차원 모델은 일반적으로 데이터 큐브로 많이 표현되는데, 이 경우 차원은 데이터 큐브를 구성하는 축(Axis)으로 생각할 수 있다. 각 축의 좌표에 해당하는 것이 차원항목이다. 각 차원을 구성하는 차원항목들의 조합에 의해 만들어지는 공간을 셀(Cell) 또는 서브큐브(Subcube)라 하며 데이터가 저장되는 공간이다[2,7]. 물론 4차원 이상인 경우 시각적으로 표현하기는 어려우며, 현실적으로 구성되는 다차원 모델은 대부분 4차원에서 8차원 이내가 일반적이다[1]. <그림 1>에서는 나이, 직업, 차종으로 구성한 데이터 큐브와 세 차원이 이루는 서브큐브의 예를 보여주고 있다[2].



<그림 1> 데이터 큐브

3. 제안하는 알고리즘

본 알고리즘은 요약된 데이터를 가지고 있는 데이터 큐브를 대상으로 최소지지도를 만족하는 모든 빈발 서브큐브를 찾는다. 각 서브큐브에는 "count"라는 정보가 있고 이것이 지지도의 개념을 대신한다. 가장 많은 차원으로 구성된 빈발 서브큐브를 최대 빈발 서브큐브라고 하면 최대 빈발 서브큐브의 모든 부분집합은 빈발이다.

예를 들면, 3차원 데이터 큐브에서는 3차원으로 이루어진 빈발 서브큐브가 최대 빈발 서브큐브이고, 8차원 데이터 큐브가 주어지면 8차원으로 이루어진 빈발 서브큐브가 최대 빈발 서브큐브가 된다.

3.1 기호 및 용어 설명

도식적인 표현을 위해 편의상 3차원으로 가정한다.

I, J, K, …… : 차원(Dimension)

$i_1, i_2, \cdots, j_1, j_2, \cdots, k_1, k_2, \cdots\cdots$: 차원항목(Member 또는 Element)

A : 큐브를 구성하는 차원의 수

Cube(I, J, K) : 차원 I, J, K로 구성한 큐브

Subcube(i, j, k) : 차원항목 i, j, k로 구성한 서브큐브

l : I 차원(x축)의 차원항목 수 , m : J 차원(y축)의 차원항목 수

n : K 차원(z축)의 차원항목 수 , B : 서브큐브의 개수($l \times m \times n$)

Minimum support(Minsup.) : 최소지지도 , Minimum confidence(Minconf.) : 최소신뢰도

Dimension columns : 차원과 차원항목 정보 , Aggregate columns : 데이터 요약 정보

a-서브큐브 : a개의 차원들로 구성된 서브큐브 , a-부분집합 : a개의 원소가 있는 부분집합

빈발 서브큐브 : 최소지지도를 만족하는 서브큐브

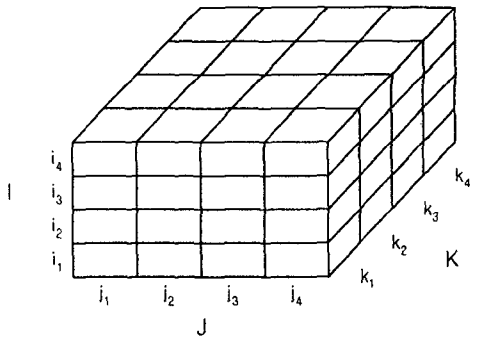
최대 빈발 서브큐브 : 빈발 A-서브큐브

(주어진 데이터 큐브의 차원의 수와 같은 개수의 차원으로 이루어진 빈발 서브큐브)

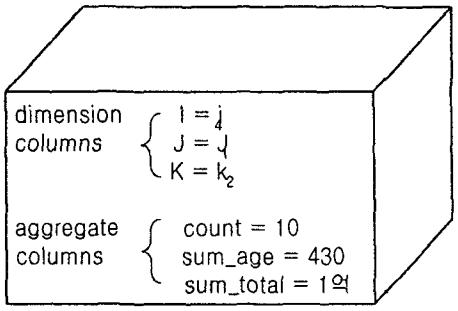
L_a^s : 빈발 a-서브큐브들의 집합

3.2 알고리즘 설명

트랜잭션 데이터베이스를 근거로 해서 요약한 데이터로 이루어진 데이터 큐브가 만들어져 있을 때, 그 큐브는 기존의 연관규칙 발견 및 장바구니 분석(Market Basket Analysis)에서 처럼 항목만으로(item-oriented) 차원을 구성할 수도 있고, 고객의 속성과 고객이 구입한 항목을 같이 차원으로 구성할 수도 있으며, 고객의 속성(customer-oriented)만으로 차원을 고려하여 구성할 수도 있다. 제시하는 알고리즘에 대한 이해를 돕기 위하여 <그림 2>와 <그림 3>을 참조한다.



<그림 2> 일반화한 데이터 큐브



<그림 3> 서브큐브의 예

다음은 알고리즘의 전개를 보여준다. 알고리즘의 전개 과정은 주어진 큐브의 모든 차원으로 이루어진 빈발 서브큐브들을 찾을 수 있는 경우(정해진 최소지지도를 만족하는 최대 빈발 서브큐브가 있는 경우)와 주어진 큐브에서 최소지지도를 만족하는 최대 빈발 서브큐브를 찾지 못하는 경우를 따로 고려한다. 이처럼 두 경우로 나누는 기준은 주어진 데이터 큐브의 A-서

브큐브들 중에서 하나라도 최소지지도를 만족하는지 아닌 지의 여부를 판단하는 것이다. 즉, 데이터 큐브의 A-서브큐브들 중에서 최소지지도를 만족하는 서브큐브가 하나라도 존재하면 이는 최대 빈발 서브큐브가 있는 경우가 되는 것이다.

3.2.1 최대 빈발 서브큐브(Maximal Large Subcube)를 찾을 수 있는 경우

1) 데이터 큐브가 구성되어 있다고 가정한다.

☞ Set Cube(I, J, K, ……)

2) 최소지지도를 만족하는 서브큐브만 선택한다. 여기서는 지정한 최소지지도를 aggregate columns에 있는 "count"와 비교하여 "count≥Minsup."인 서브큐브를 찾는다. 선택된 빈발 서브큐브의 차원의 수가 주어진 데이터 큐브의 차원의 수와 같으면 이러한 빈발 서브큐브를 최대 빈발 서브큐브라 한다.

3) 구해진 최대 빈발 서브큐브의 dimension columns과 aggregate columns의 내용을 평면 테이블의 형태로 바꾸어 표현한다.(3차원은 상관없지만 4차원 이상이면 시각적으로 볼 수 없으므로 평면 테이블화하는 것이 유리하고 입체 큐브와 구분하기 위해 평면 테이블이라는 용어를 사용했다.)

4) "최대 빈발 서브큐브의 a-부분집합은 모두 빈발이다.(a≤A)"는 명제를 사용한다. (A-1)차원으로 구성되는 빈발 서브큐브를 찾는다. 어떤 차원을 제거할 것인가에 대한 문제는 두 가지 방법으로 해결할 수 있다.

① 사용자가 임의로 관심 없는 하나의 차원을 선택하여 제거할 수 있을 것이다. 항목 제약이 있는 연관규칙 문제에서처럼 관심있는 차원만을 고려하는 방법으로서, (A-1)개 차원으로 이루어진 서브큐브들을 대상으로 해서 제거한 나머지 차원에 대해 "count"를 합산한다. 만약, $\{I=i_1, J=j_2, K=k_4\}$가 최대 빈발 서브큐브이고 세 차원에서 K차원을 뺐다면 $\{I=i_1, J=j_2, K=all\}$의 "count"를 더한다. $\{I=i_1, J=j_2, K=all\}$의 의미는 $I=i_1$, $J=j_2$이면서 K차원에 대해서는 모든 차원항목을 고려한다는 뜻이다. 각각의 최대 빈발 서브큐브에 대해 차원을 하나씩 감소시키면서 1-부분집합까지 이 과정을 반복한다. 물론 한 번에 두 개 이상의 차원을 빼는 방법도 가능하다. 이 방법은 특정 차원만을 제거하기 때문에 모든 빈발 서브큐브를 구할 수가 없고, 사용자 또는 의사결정자에 따라서 마이닝 결과가 달라지므로 "Partial mining association rules"라고 한다.

② 또 다른 방법은 최대 빈발 서브큐브 각각에 대하여 제거 가능한 모든 차원을 고려하는 것이다. 만약, $\{I=i_1, J=j_2, K=k_4\}$가 최대 빈발 서브큐브이면 차례로 I, J, K 차원을 제거해 가면서 모든 경우의 수를 따지는 것이다. 즉, $\{I=all, J=j_2, K=k_4\}$, $\{I=i_1, J=all, K=k_4\}$, $\{I=i_1, J=j_2, K=all\}$의 "count"를 모두 계산하여 2-부분집합을 찾는다. 각각의 최대 빈발 서브큐브에 대해 차원을 하나씩 감소시키면서 1-부분집합까지 이 과정을 반복한다. 따라서 이 방법은 주어진 큐브에서 구할 수 있는 모든 빈발 서브큐브를 구하는 것이므로 "Full mining association rules"라 한다.

5) 구해진 빈발 서브큐브(L_a^s)를 1.1절에서 설명한 연관규칙 발견 공식에 적용하여 신뢰도를 구하고 그 값이 최소신뢰도보다 크거나 같으면 강한 연관규칙이라고 한다.

3.2.2 최대 빈발 서브큐브(Maximal Large Subcube)를 찾을 수 없는 경우

구성된 큐브의 모든 A-서브큐브가 주어진 최소지지도를 만족하지 못하는 경우이다. 여기서는 세 가지 방법을 제시한다.

1) 사용자가 최소지지도를 더 낮게 정한다.

2) 다른 방법은 차원을 감소시키는 방법으로서 위의 3.2.1절에서 설명한 알고리즘 전개 과정의 4)번 과정을 따르는 것이다. 차원을 감소시킨다는 의미는 지지도(즉, "count")를 높인다는 것이다.

3) 나머지 방법은 차원항목을 결합하는 것이다. 각 차원은 차원항목들로 구성되어 있어서 각 차원에 대한 차원항목의 결합은 지지도를 높이게 되는데, 이를 "차원항목의 일반화(generalization)"라고 한다. 예를 들어, 나이라는 차원이 20대, 30대, 40대, 50대의 네 개의 차원항목으로 이루어져 있다면 20~30대와 40~50대로 결합할 수 있다. 즉, 차원항목의 수를 4개에서 2개로 줄이는 것이다. 또 차종을 차원으로 갖는 큐브에서 p, q, r, s, t, u라는 자동차 브랜드가 차원항목이면 소형차(p, q), 중형차(r, s), 대형차(t, u)라는 세 개의 차원항목으로 일반화시킬 수 있다. 이러한 일반화는 서브큐브의 수를 감소시키고 차원의 서로 다른 계층사이의 연관규칙 발견을 가능하게 한다.

4. 예 제

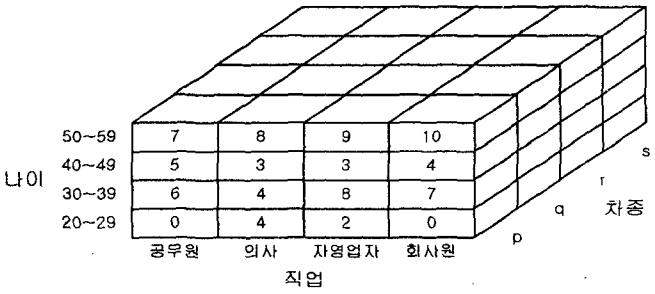
4.1 예제 1

<표 1>은 서울지역의 한양자동차 회사의 트랜잭션 데이터베이스이다. 기존의 장바구니 분석처럼 항목끼리의 관계를 파악하는 것이 아니라 고객의 속성을 포함한 연관관계를 마이닝하여 고객 세분화를 통한 DB 마케팅에 이용하고 일대일 마케팅을 실현하여 자사의 기존 고객 유지 및 신규 고객 창출에 응용한다.

<표 1> 트랜잭션 데이터베이스

번호	이름	직업	나이	차종	취미	……
1	…	공무원	52	p	…	……
2	…	회사원	26	q	…	……
3	…	자영업자	45	r	…	……
4	…	공무원	57	r	…	……
⋮	⋮	⋮	⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮	⋮	⋮	⋮

위의 데이터를 아래 <그림 4>와 같이 데이터 큐브 형태로 요약해 놓았다고 하자.



<그림 4> 자동차 회사의 데이터 큐브

위와 같은 형태로 4×4×4=64개의 서브큐브가 구현되어 있고 총 고객의 수는 300명이라고 가정하자. 최소지지도를 3%로 정하면 300×0.03=9명 이상이면 빈발이 된다. <그림 4>의 3차원 데이터 큐브에서는 최소지지도(9명)보다 크거나 같은 지지도를 갖는 서브큐브들이 존재하므로 이 예제는 최대 빈발 서브큐브를 찾을 수 있는 경우이다. 최대 빈발 서브큐브만을 골라서 평면 테이블로 바꾼 형태가 <표 2>에 있고 계속해서 차원을 감소시키면서 빈발 2-서브큐브를 찾아나간 결과를 <표 3>에서 보여준다. 계속해서 3.2.1절의 알고리즘의 4)번 과정에 의해 빈발 1-서브큐브를 찾을 때까지 실행한다. <표 3>에서 "all"의 의미는 해당 차원의 세부적인 차원항목을 고려하지 않는다는 뜻이다. 즉 지지도가 29명인 {공무원, 50~59, all}은 직업이 공무원이면서 나이가 50대인 고객을 의미하며 차종은 상관없다(모든 차종을 포함한다)는 뜻이다.

<표 2> 평면 테이블-최대 빈발 서브큐브(L_3^s)

직업	공무원	의사	자영업자	회사원	……
나이	50~59	30~39	50~59	50~59	……
차종	r	r	p	p	……
지지도	9	11	9	10	……

<표 3> 빈발 2-서브큐브(L_2^s)

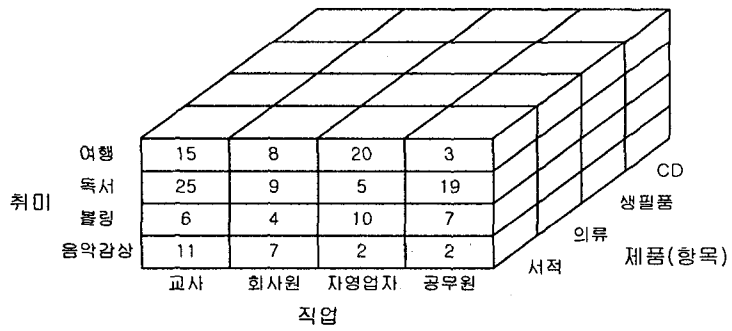
직업	공무원	all	자영업자	회사원	……
나이	50~59	30~39	all	50~59	……
차종	all	r	p	all	……
지지도	29	30	24	18	……

이 예제는 최대 빈발 서브큐브를 찾을 수 있는 경우인데, 최소신뢰도가 50%이면 얻어지는 연관규칙의 예는 {공무원, 50~59}⇒{r}은 9/29=0.31(신뢰도)이고 {회사원, 50~59}⇒{p}은 10/18=0.56(56%)이다. 두 번째 연관규칙의 신뢰도(56%)는 주어진 최소신뢰도(50%)를 만족하므로 강한 연관규칙이다.

일반적으로 연관규칙의 종류는 useful, trivial, inexplicable로 나누는데[2], useful은 기존에 발견하지 못한 고품질이면서 활용가능한 정보이고 trivial은 그 분야의 사업과 관련해서 이미 알려져 있는 규칙이고 inexplicable은 항목이나 고객의 성향과는 아무런 상관이 없는 규칙을 말한다. "의사이면 30대이다"({의사}⇒{30~39})라든지 "20대면 여자이다"({20~29}⇒{여자})와 같은 연관규칙은 유용(useful)한 연관규칙이 아니다. 비록 빈발이지만 고객의 속성끼리의 연관성은 의미가 없을 수도 있어서 주의해야 한다. 다음의 예제 2와 예제 3에서는 빈발 서브큐브를 찾는 과정은 생략하고, 발견한 빈발 서브큐브를 이용하여 생성한 연관규칙을 마케팅에 응용하는 사례를 보여준다.

4.2 예제 2

최근 각광받고 있는 인터넷 쇼핑몰 사업은 대부분 회원 가입을 요구하며 사용자들은 자신의 속성, 즉 고객 정보를 입력한다. 아래의 <그림 5>는 어떤 인터넷 쇼핑몰 업체의 고객에 대한 정보를 요약해 놓은 큐브이다. 위의 예제 1과 같은 방법으로 빈발 서브큐브를 찾은 후에 얻어진 연관규칙은 다음과 같을 수 있다. 만약 {교사, 여행}⇒{의류}가 70%의 신뢰도를 가진 강한 연관규칙이라고 하자. 이 규칙은 직업이 교사이면서 여행을 좋아하는 고객에게 "의류"의 신제품이 나왔을 때나, 신규 고객 창출의 관점에서 잠재 고객(교사)에게 구매할 수 있도록 텔레마케팅(Tele-Marketing)에 이용된다.



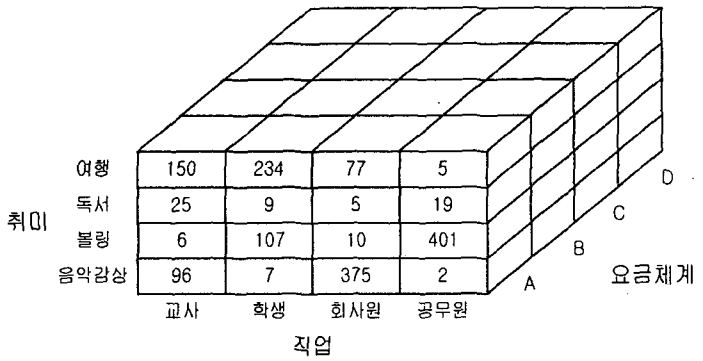
<그림 5> 인터넷 쇼핑몰 업체의 데이터 큐브

이러한 일대일 마케팅은 요즘의 마케팅 흐름이 Mass Marketing에서 Segmentation Marketing을 거쳐 Individual Marketing으로 옮겨가는 추세에 부합한다.

위의 큐브가 "카드종류"라는 차원을 포함하여 4개의 차원으로 구성되어 있다면 {여행, 교사, 생필품}⇒{A신용카드} 또는 {CD, B신용카드, 회사원}⇒{볼링}와 같은 연관규칙이 나올 수도 있을 것이다. 이런 경우, 취미가 여행이고 생필품을 현금으로 구입하는 교사에게 A신용카드 가입을 권유하거나 회사원이 B신용카드로 음악 CD를 구입하면 볼링 할인권을 제공하는 등의 마케팅을 할 수 있다. 즉, 신용카드 회사, 볼링장 업체, 레코드 판매점의 세 개의 회사가 연관규칙을 공유하여 교차 판매(Cross-Selling)를 할 수도 있다.

4.3 예제 3

<그림 6>은 휴대용 전화를 판매하는 어느 이동통신 회사의 고객 정보를 담은 데이터 큐브이다. 이 데이터 큐브는 휴대전화기라는 항목(item)은 차원화하지 않고 그 항목을 구입한 고객들의 속성으로만 차원을 고려해서 만들어졌다. 이 데이터 큐브를 대상으로 알고리즘을 실행해서 {교사, 요금체계 'A'}⇒{음악감상}와 같은 연관규칙을 발견했다면, 이 회사는 이러한 연관규칙들을 고객 세분화를 통한 Target Marketing에 유용하게 사용한다.



<그림 6> 이동통신 회사의 데이터 큐브

<그림 6>의 데이터 큐브에 "구독신문"이라는 고객의 속성을 포함한 4차원 큐브를 구성하여 알고리즘을 실행한다. {음악감상, 요금체계 'B', 학생}⇒{한양일보}와 같은 연관규칙이 발견되면 이 정보를 가지고서 이동통신 회사가 아니라 신문사가 Target Marketing을 할 수도 있을 것이다. 즉, 이종 업종간의 정보 공유를 통한 Cross Marketing도 마케팅 방법의 하나이다.

5. 결 론

연관규칙 발견은 대용량 데이터베이스에서 감추어져 있긴 하나 잠재적 사용가치가 큰 항목들 사이의 패턴이나 추세 및 흥미로운 규칙들을 발견하는 데이터 마이닝 기법이다. 최근에 데이터 웨어하우스의 효과적 활용방안에 대한 관심이 집중되면서 OLAP는 데이터 마이닝과 상호 보완관계를 형성하며 발전하고 있지만, 데이터 마이닝과 OLAP 기술을 통합하려는 연구에 대해서 OLAP가 다양한 마이닝 기법들과 연동하는 알고리즘은 적었다.

기존에는 연관규칙 발견은 트랜잭션 데이터베이스를 대상으로 하고, 빠른 응답시간을 제공하는 OLAP는 요약된 데이터 큐브를 대상으로 보았다. 본 논문에서는 만들어진 데이터 큐브를 연관규칙 발견에 이용하는 알고리즘을 제시했다. 큐브에 대해서 최소지지도를 입력하여 최대 빈발 서브큐브를 찾아, 그것의 부분집합들을 빈발로 보고 빈발 서브큐브들을 찾아서 연관규칙을 마이닝하여 고객 세분화를 통한 데이터베이스 마케팅에 응용한다. 추출된 연관규칙들은 빠른 의사결정과 효율적인 고객 마케팅을 제공하여 기업들이 심화된 경쟁속에서 차별화된 전략으로 경쟁력을 강화할 수 있도록 한다.

참고문헌

[1] 조재희, 박성진, OLAP 테크놀로지, SIGMA CONSULTING GROUP, SEOUL, 1999.
[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] D. Cheung, J. Han, V. Ng, and C. Y. Wong, Maintenance of Discovered Association Rules in Large Databases: An Incremental Updating Technique, Proc. of 1996 Int'l Conf. on Data Engineering (ICDE'96), New Orleans, Louisiana, USA, Feb. 1996.
[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] Harinarayan, V., Rajaraman, A., Ullman, J., Implementing Data Cubes Efficiently, In Proc. of ACM SIGMOD, pp. 205-216, Montreal, Canada, June 1996.
[8] 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.
[9] 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.
[10] M. Klemettinen, H. Mannila, P. Ronkainen, H. Toivonen, and A. I. Verkamo, Finding interesting rules from large sets of discovered association rules, Proc. 3rd Int'l Conf. on Information and Knowledge Management, Gaithersberg, Maryland, pp. 401-408, Nov. 1994.
[11] 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.
[12] 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.
[13] 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.
[14] R. Srikant and R. Agrawal, Mining Generalized Association Rules, Proc. of the 21st Int'l Conference on Very Large Databases, Zurich, Switzerland, Sep. 1995.
[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.
[17] Theodoratos, D., Sellis, T., Data Warehouse Configuration, In Proc. of the 23th International Conference on VLDB, pp. 126-135, Athens, Greece, August 1997.