□ 기술해설 □

연관 규칙 탐사와 그 응용

성신여자대학교 박종수*·유원경**·홍기형**

1. 서 론

컴퓨터 시스템의 발달과 데이타베이스 시스템의 사용의 증가로 컴퓨터에 저장되는 데이타의 양은 폭발적으로 증가하고 있다. 현재 컴퓨터에 저장되어 있는 대용량 데이타베이스에는 사용자가 미처 파악하지 못하는 중요한 정보가 포함되어 있을 수 있다. 이것은 데이타베이스 스키마를 작성할 때에 이미 잘 알려진 내용의 통계적인 자료 또는 원하는 정보를 빠르게 검색하기 위해서 스키마를 설계하였기 때문에 일정한 형식의 질의(SQL type)를 벗어난 정보 검색에는 대응하지 못하는 데이타베이스 시스템의 한계를 나타낸다. 데이타베이스 분야와 인공지능의 지식 발견 분야의 결합적인 접근 방식으로 대용량 데이타베이스에서 쉽게 찾아낼 수 없고 알려져 있지 않은 일정한 패턴 또는 정보를 찾아내려는 시도가 1990년대의 초기부터 이루어지고 있다. 데이타 마이닝(Data Mining)[1]은 대용량의 데이타에서 숨겨진 유용한 패턴을 추출하는 방법론을 일컫고, 이 것은 OLAP과 Data Warehousing을 구축할 때 중요한 도구로 사용되고 있다[2].

데이타에서 숨겨진 패턴을 탐사하는 연구중에서 연관 규칙 탐사[3]가 가장 많은 연구가 이루어졌고 그 결과인 알고리즘의 적용으로 새로운 패턴을 찾아내고 있다. 연관 규칙은 한 항목들의 그룹과 다른 항목들의 그룹사이에 강한 연관성이 있음을 밝혀준다. 예를 들면, 소매점에서 목요일에 기저귀를 구매하는 고객들은 맥주도 동시에 구매한다는 연관성이 있음을 연관 규칙 탐사에서 알아냈다. 이제까지 규칙이라 함은 물리학이나 수학에서처럼 거의 100% 정확한 규칙을 생각하게 된다. 그러나, 연관 규칙은 100% 정확한 규칙이 아니라 데이타베이스에서 사용자가 지정하는 정도의 규칙을 발견해낸다. 소매점에서 각 고객이 구매하는 물품들의 집합을 한 트랜잭션이라 하고, 이런 트랜잭션들을 일정한 기간동안 저장한 것을 데이타베이스라 하면, 앞에서 언급한 기저귀를 사는 사람은 맥주를 구매한다는 것을 규칙으로 표현하면 다음과 같다 : "기저귀 맥주[10% of support, 80% of confidence]". 여기서 10%의 지지도라는 것은 주어진 데이타베이스의 트랜잭션들(고객들)중에서 10%가 기저귀와 맥주를 동시에 산다는 것이고, 80%의 신뢰도라는 것은 기저귀를 사는 고객들 중에서 80%가 맥주를 산다는 것이다. 연관 규칙 탐사에서는 사용자가 지지도와 신뢰도의 값을 적절하게 입력하여 이미 발생한 트랜잭션들에서 물품들 상호간의 연관성을 발견해낼 수 있고, 이를 바탕으로 고객들의 구매 패턴을 알 수 있고 더불어 시장성 예측 등에 그대로 적용할 수 있다. 이와 같이 저장되어 있는 대용량의 데이타에서 잘 알려져 있지 않고 찾아내기 어려운 유용한 패턴을 탐사한다는 것은 어떤 의사결정 시스템에서도 아주 큰 역할을 할 수 있을 것이다.

본 논문에서는 연관 규칙에 대한 정의를 알아보고 이를 탐사하는 방법론에 대하여 설명한다. 그리고 연관 규칙에서 시간 개념이 포함된 순차 패턴과 world wide web 등에서 사용자가 접근하는 패턴에 관한 규칙 탐사인 순회 패

*정 회 원
**종신회원

턴 탐사에 대해서도 설명한다.

2. 연관 규칙

2.1 빈발 항목집합의 정의

트랜잭션이 빈번하게 발생하는 소매점의 물품 판매에서 만들어지는 트랜잭션 데이타베이스[3, 4]를 고려해 보자. 항목들(예를 들면, 소매점에서 판매된 물품 항목들)의 집합 $I=\{i_1, i_2, \cdots, i_m\}$이 주어지면, 트랜잭션 T는 I의 부분집합으로 정의된다($T \subseteq I$). 트랜잭션이라 함은 구매자가 소매점에서 한꺼번에 구매하는 물품들의 집합으로 볼 수 있다. 집합과 같이 트랜잭션들은 중복된 항목을 허용하지 않는다. 그러나 우리는 순수한 집합의 개념을 확장하고 트랜잭션과 다른 모든 항목집합들 내에 있는 항목들은 정렬된 것으로 가정한다. 데이타베이스 D를 n개의 트랜잭션들의 집합이라 하고 각 트랜잭션은 고유한 트랜잭션 번호(TID)가 부여된다. 만일 트랜잭션 T가 X의 모든 항목들을 포함한다면($X \subseteq T$), T가 집합 X(물론, $X \subseteq I$)를 지지한다(support)고 한다. 우리는 X의 지지도를 생략된 형태 supp(X)로 정의하며 이는 X를 지지하는 D에 있는 모든 트랜잭션들의 개수를 의미한다. 만일 사용자가 정한 최소 지지도 s_{min}에 대하여 $supp(X) \geq s_{min}$이라면, 집합 X는 빈발하다(이런 경우에 항목들의 집합 X를 일반적으로 large itemset이라 하고 또는 frequent itemset이라고도 한다)고 한다. 최소 지지도를 사용하는 이유는 D에 대하여 관심있을 정도로 빈발하게 나타나는 항목만을 고려하기 위함이다. 항목집합 X의 개수를 $k=|X|$로 나타내고 이를 k-항목집합이라 부른다.

다음은 빈발 항목집합을 효과적으로 찾는 과정[4, 5, 6]에서 얻어진 특성들로 대부분의 연관 규칙 탐사 알고리즘에서 사용되고 있다.

• 특성 1(부분집합의 지지도) : 만일 항목집합 A, B에 대하여 $A \subseteq B$이면, B를 지지하는 D의 모든 트랜잭션들이 필연적으로 A 또한 지지하므로 $supp(A) \geq supp(B)$이다.

• 특성 2(빈발하지 않은 집합들의 상위집합들(supersets)은 빈발하지 않다) : 만일 항목집합 A가 D에서 최소 지지도에 미치지 못한다면, 즉 $supp(A) < s_{min}$, 특성 1에 의하여 $supp(B) \leq supp(A) < s_{min}$이기 때문에 A의 모든 상위집합 B는 빈발하지 않을 것이다.

• 특성 3(빈발 항목집합들의 부분집합들은 빈발하다) : 항목집합 B가 D에서 빈발하다면, 즉 $supp(B) \geq s_{min}$, 특성 1에 의하여 $supp(A) \geq supp(B) \geq s_{min}$이므로 B의 모든 부분집합 A는 D에서 또한 빈발할 것이다. 특히, 만약 $A=\{i_1, i_2, \cdots, i_k\}$가 빈발하면, 그것의 모든 k개의 (k-1)-부분집합들도 빈발하다. 그 역은 성립하지 않는다.

2.2 연관 규칙의 정의

X와 Y를 항목들의 집합이라 하자. 연관 규칙(association rule)은 $R : X \rightarrow Y$ 형식의 함축이고, 이때 X와 Y는 서로 같은 원소를 갖지 않는 항목집합이다 : $X, Y \subseteq I$이고 $X \cap Y = \phi$이고, $Y \neq \phi$여야 한다. X를 규칙의 조건부(antecedent)라 하고 Y를 결과부(consequent)라 한다. 만일 한 트랜잭션이 X를 지지한다면, 또한 어떤 확률에 의해 Y도 지지할 것이라는 예측으로 이해될 수 있는 것이 연관 규칙이다. 이런 확률을 이 규칙의 신뢰도(conf(R)로 표시)라 한다. R의 신뢰도는 X를 지지하는 T에 대하여 Y또한 지지할 조건부 확률로 정의된다. 즉, $conf(R) = supp(X \cup Y)/supp(X)$. D에 있는 규칙 R에 대한 지지도는 $supp(X \cup Y)$로 정의한다. 규칙의 신뢰도는 얼마나 조건부에 대하여 결과부가 자주 적용할 수 있는지를 나타내고 반면 지지도는 그 규칙 전부가 얼마나 믿을 만한지를 보여준다. 규칙이 데이타베이스에서 적절해지려면 충분한 지지도와 신뢰도를 가져야 한다. 그러므로 어떤 주어진 최소 신뢰도 c_{min}과 최소 지지도 s_{min}에 대하여 만일 $conf(R) \geq c_{min}$이고 $supp(R) \geq s_{min}$하면 규칙 R은 D에 대하여 성립한다. 연관 규칙에 대하여 추가적인 몇가지 특성에 대한 설명은 [6]에서 참조할 수 있다.

3. 연관 규칙 탐사의 기본적인 접근방식

연관 규칙에 대한 정의에 이어서 우리는 기

본적인 규칙 탐사 알고리즘의 구조를 묘사할 수 있다. 연구된 알고리즘들이 서로 다름에도 불구하고 그들 모두는 기본적인 스키마를 사용한다. 이들 요소는 서로 다르게 배열될 수 있고 전 스키마가 반복적으로 적용될 수도 있다. 주어진 데이타베이스에서 탐사되는 연관 규칙이 사용자가 정의한 최소 지지도와 최소 신뢰도 이상의 값들을 가져야 하므로, 연관 규칙을 탐사하는 문제는 기본적으로 다음의 두 단계로 구성된다[4]:

• 단계 1: 빈발 항목집합들(large itemsets)을 찾아낸다. 미리 결정된 최소 지지도 s_{min} 이상의 트랜잭션 지지도를 가지는 항목집합들의 모든 집합들을 빈발 항목집합들이라 한다. 그 외 모든 항목집합들은 작은 항목집합들(small itemsets)이라 한다.

• 단계 2: 데이타베이스로 부터 연관 규칙을 생성하기 위하여 빈발 항목집합을 사용한다. 모든 빈발 항목집합 L에 대해서 L의 모든 공집합이 아닌 부분집합들을 찾는다. 각각의 그러한 부분집합 A에 대하여, 만약 supp(A)에 대한 supp(L)의 비율이 적어도 최소 신뢰도 c_{min} 이상이면($supp(L)/supp(A) \geq c_{min}$), $A \rightarrow (L-A)$의 형태의 규칙을 출력한다. 이 규칙의 지지도는 supp(L)이고 신뢰도는 supp(L)/supp(A)이다.

연관 규칙 탐사의 전체 성능은 사실 첫번째 단계에서 결정된다. 먼저 빈발 항목집합을 확인한 후에 해당되는 연관 규칙을 단계 2의 방법으로 쉽게 유도할 수 있다.

3.1 빈발 항목집합 찾기

잠재적인 빈발 항목집합들의 수는 모든 항목들의 멱집합(power set)의 크기와 같다. 이것은 고려될 항목들의 크기에 대하여 기하급수적으로 증가한다. 간단한 알고리즘은 낭비적인 탐사를 하고 그것이 빈발한지의 여부를 판단하기 위해 그 멱집합에 속한 모든 집합들을 테스트할 수도 있다. 모든 알고리즘이 고려하는 기본적인 방법은 후보(candidates)라 지칭하는 빈발 가능성이 있는 항목집합들의 생성을 포함한다. 이들 후보 항목집합들 중에 실제로 빈발한 항목들을 찾기 위해서는 각 후보 항목집합들에 대한 지지도가 데이타베이스를 읽어나가면서 계산되어야 한다. 후보 항목집합의 발생 빈도를 계산하는 것은 상당량의 프로세싱 시간과 메모리를 요구하기 때문에 연관 규칙 탐사 알고리즘의 성능은 후보들의 수에 비례한다.

빈발 항목집합을 찾아내는 과정을 이해하기 위하여 간단한 데이타베이스에서 후보 항목집합의 생성과 그것에서 빈발 항목집합을 찾아내는 방법을 설명하기로 한다. 이를 위해서 지금까지 발표된 알고리즘에서 전형적인 방법론인 Apriori[4]가 적용한 방법론을 설명한다. Apriori에서는 각 패스에서 빈발 항목집합의 후보집합을 구성하고 난 후에 각 후보 항목집합의 발생 빈도수를 계산하고, 사용자가 정의한 최소 지지도를 기초로 하여 빈발 항목집합을 결정한다. 그림 1은 설명을 위한 데이타베이스이고, 그림 2는 빈발 항목집합을 찾는 과정을 설명한다. 첫번째 패스에서 각 항목의 발생 빈도수를 세기 위하여 단순히 모든 트랜잭션들을 스캔하여 읽는다. 후보 1-항목집합들의 집합 C_1은 그림 2에서와 같이 얻어진다. 최소 트랜잭션 지지도가 2라고 가정하면($s_{min}=2$), 필요로

TID	Items
100	A C D
200	B C E
300	A B C E
400	B E

그림 1 트랜잭션 데이타베이스의 예

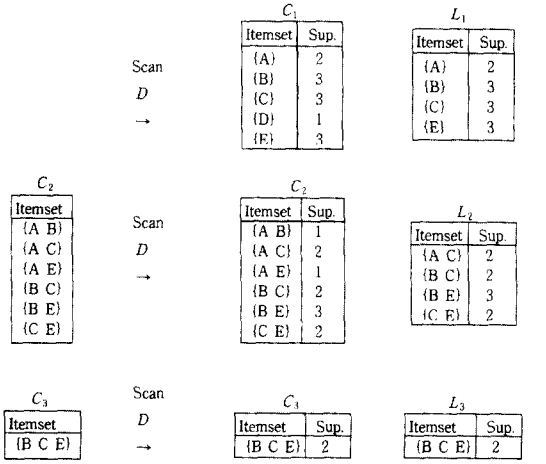


그림 2 후보 항목집합의 생성과 빈발 항목집합

하는 최소 지지도를 갖는 후보 1-항목집합들로 구성되는 빈발 1-항목집합들의 집합 L_1이 결정될 수 있다.

빈발 2-항목집합들의 집합을 탐사하기 위해서는, 모든 부분집합도 역시 최소 지지도를 가져야 한다는 사실에 입각하여 Apriori는 후보 항목집합들의 집합 C_2를 생성하기 위해 $L_1 * L_1$을 사용하였다. 여기서 *는 접합 연산자다. C_2는 $\left(\frac{|L_1|}{2}\right)$개의 2-항목집합들로 이루어진다. $|L_1|$이 크게 되면 $\left(\frac{|L_1|}{2}\right)$는 극도로 큰 숫자가 됨을 알 수 있다. 다음으로 D에 속한 네개의 트랜잭션들이 스캔되어 읽히고 C_2에 속한 각 후보 항목집합의 지지도는 계산된다. 그림 2에서 두번째 행의 가운데 테이블은 C_2에 속한 후보 항목집합의 지지도 계산결과를 나타낸다. 빈발 2-항목집합들의 집합 L_2는 C_2에 속한 각 후보 2-항목집합의 지지도에 기초하여 결정된다.

후보 항목집합들의 집합 C_3는 L_2에서 다음과 같이 생성된다. L_2에서 첫 항목이 같은 두개의 빈발 2-항목집합들을 먼저 확인한다. 예를 들면, {BC}와 {BE}에서는 B가 동일한 항목이다. 다음으로, Apriori는 {BC}와 {BE}의 두번째 항목들로 구성된 2-항목집합 {CE}가 빈발 2-항목집합들에 속하는지를 검사한다. {CE}가 L_2의 원소로 빈발 집합이므로, {BCE}의 모든 부분 집합들은 빈발하다는 것을 알았고, 그러므로 {BCE}는 후보 3-항목집합이 된다. L_2에서 더 이상의 다른 후보 3-항목집합을 구할 수 없다. 그러면, Apriori는 모든 트랜잭션들을 스캔하면서 그림 2에서와 같이 빈발 3-항목집합을 구성한다. C_3를 기본으로 하여 D를 스캔하여 L_3를 찾아낸다. L_3에서 부터 구성될 수 있는 후보 4-항목집합이 없으므로, 여기서 빈발 항목집합을 발견하는 과정을 마친다.

3.2 Aprori에서 후보 항목집합의 생성과 지지도 계산

연관 규칙의 문제는 먼저 [3]에서 제기되어 그후에 많은 연구 논문이 뒤따랐다. 연관 규칙 탐사에 관하여 발표된 논문들은 빈발 항목집합 생성과 규칙 형성을 분리하여, 주로 전자에 많은 관심이 집중되어 여러 알고리즘들에 의해 다양한 방법으로 문제 해결을 시도하였다. 이 절에서는 연관 규칙 탐사 알고리즘 중에서 전형적인 알고리즘이 Apriori[4]이다.

AIS[3]에서의 많은 수의 후보집합의 생성은 Apriori-gen이라는 새로운 후보 항목집합의 생성 전략을 개발케 하였으며 이는 Aproiri가 연관 규칙 탐사 부분에 기여한 중요한 부분이다. Apriori-gen은 후보 항목집합의 수를 줄이는데 성공적이어서 그 이후 대부분의 알고리즘에서 사용하게 되었다. 이 방법은 조인(join) 단계와 전지(prune) 단계로 구성된다. 후보 (k+1)-항목집합은 단지 모든 k-부분집합이 빈발할 때만 선택되어질 것이다. 다음 알고리즘에서의 코드에서 보여진 것처럼, Apriori-gen은 빈발 항목집합 L_k를 입력으로 사용하고 그들의 k-1개의 같은 항목들을 갖는 쌍 a와 b를 찾는다. k-1개의 공통된 항목들과 두개의 다른 항목들을 찾아서 후보 (k+1)-항목집합을 형성하기 위하여 조인된다.

```
Algorithm Apriori-gen
insert into C_{k+1}                {join step}
select a.item_1, a.item_2, ..., a.item_k, b.item_k
from L_k a, L_k b
where a.item_1 = b.item_1, ..., a.item_{k-1} =
        b.item_{k-1}, a.item_k < b.item_k
{prune step: now prune rules with subsets missing in L_k}
forall itemset c ∈ C_{k+1} do
  forall k-subsets s of c do
      if (s ∉ L_k) then delete c from C_{k+1}
```

두번째 단계에서는 만들어진 후보 (k+1)-항목집합의 부분집합 k-항목집합들이 이미 L_k 예에 있는지를 검사하여 없으면 이 후보항목집합을 전지한다. 예를 들어 {1, 3, 4, 6}과 {1, 3, 4, 8}이 빈발하다면 두 항목집합은 조인되어 후보 항목집합 {1, 3, 4, 6, 8}을 생성한다. 확인되어져야 할 부분집합은 {3, 4, 6, 8}, {1, 4, 6, 8}, {1, 3, 6, 8}이며 만일 이들이 빈발하지 않다면 버려져야 한다.

Apriori 알고리즘은 그 각각의 패스가 주어진 항목 개수를 가지는 모든 후보들을 생성하

는 Apriori-gen에 대한 호출과 이 후보들에 대한 지지도를 계산하는 카운팅 단계로 구성되어있다. 지지도 계산 단계에서는 전체 데이타베이스를 스캔한다. Apriori 알고리즘은 k번째 패스의 지지도 계산 단계에 있는 트랜잭션 T를 읽을 때 트랜잭션 T에 의해 지지되는 모든 k-항목후보들의 지지도 카운터를 증가시켜야 한다. 지지도 계산을 능률적으로 수행하기 위해서 Apriori알고리즘은 해쉬 트리에 후보 항목집합들을 저장한다.

3.3 기존의 알고리즘들

연관 규칙 탐사 알고리즘으로 AIS[3]가 발표되었고, SETM[7]은 표준적인 데이타베이스 연산을 사용하여 빈발 항목집합들을 구하였다. 앞 절에서와 같이 Apriori[4]와 [5]에서 후보 항목집합을 효과적으로 구하는 방법이 동시에 연구되어 발표되었다. DHP[8]는 Apriori와 비교하여 후보 2-항목집합들을 작게 효율적으로 구하는 방법과 이것에 기초로 전체 트랜잭션의 크기와 개수를 줄여나가는 방법을 제시하였다. PARTITION[9]은 입/출력 횟수를 줄이는 방안을 제안하였고, 전체 데이타베이스 샘플링을 취하여 탐사 시간을 줄이는 방법이 [10]과 [11]에서 제안하였다. DIC[12]는 기존의 방법보다 더 작은 패스 횟수로 빈발 항목집합들을 찾아낼 수 있는 자료구조와 규칙의 유용성 측정 방법에서 확신도(conviction)에 기초로 한 함축 규칙(implication rule)을 제시하였다.

최근에는 일정 주기 상에서 나타나는 cyclic association rule을 찾는 방법[13]과 특정 물품들 간에는 연관성이 없다는 것을 표시해주는 negative association을 탐사하는 방법[14] 등이 연구되고 있다. 그리고, 탐사하는 연관 규칙의 조건부나 결과부에 제한을 준 상황에서 빠르게 연관 규칙을 찾아내는 방법[15]도 연구되고 있다.

4. 연관 규칙 탐사의 응용

4.1 일반화된 연관 규칙

각 항목(물품)은 어떤 분류(taxonomy) 기준에 의하여 한 분류에 속한다. 연관 규칙 탐사에서도 이런 분류를 사용하여 보다 포괄적인 의미를 갖는 규칙을 찾아내는 연구가 수행되었다. 일반화된 연관 규칙(Generalized Association Rules)[16]의 탐사 문제에서는 각 항목의 분류를 포함하는 연관성을 찾아내게 하였다.

이런 경우에 트랜잭션이 항목들로 이루어진다면 각 항목의 분류도 연관성 탐사에 포함된다. 그런 분류(taxonomy)의 한 예가 그림 3에 보여진다; 이 taxonomy는 Jacket is-a Outerwear, Ski Pants is-a Outerwear, Outerwear is-a Clothes이다. 사용자들은 서로 다른 taxonomy 수준들을 걸쳐놓게 하는 규칙의 생성에 관심이 있다. 예를 들어 고객이 Jacket과 Hiking Boots를 함께 사고 Ski Pants를 Hiking Boots와 함께 샀다는 사실로부터 "Outerwear를 산 사람은 Hiking Boots를 사는 경향이 있다"라는 규칙을 추론할 수 있다. 그러나 "Outerwear→Hiking Boots"라는 규칙에 대한 지지도는 "Jacket→Hiking Boots"와 "Ski Pants→Hiking Boots"의 지지도의 합이 아닐수도 있다. 왜냐하면 어떤 고객들은 Jacket, Ski Pants, HikingBoots를 같은 트랜잭션에서 동시에 구입할 수도 있기 때문이다.

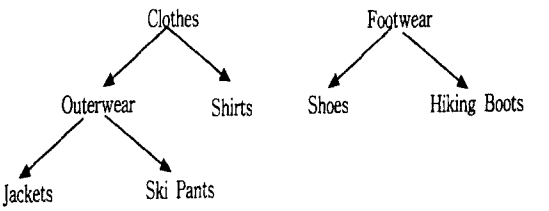


그림 3 Taxonomy의 예

서로 다른 분류 수준에 상응하는 연관성을 찾는 것이 기본적이고, 보다 상위의 수준에 해당하는 일반적인 연관 규칙을 탐사한다. 분류상 하위 수준의 연관 규칙은 예상되는 상대적인 관심도(R-interesting)미만의 지지도를 갖는 규칙은 제거되고 그 이상의 지지도를 갖는 규칙에 대해서는 계속해서 찾아낸다. 실제 데이타에서는 생성된 규칙들의 약 50%가 제거되었다.

4.2 순차 패턴

순차 패턴(Sequential Pattern) 탐사[17, 18]는 한 트랜잭션 안에서 발생하는 항목들간의 연관 규칙에 시간의 변이를 추가한 것이다. 즉 연관 규칙은 트랜잭션 안에서 어떤 항목을 함께 사는가 하는 문제로 트랜잭션 내의 문제인 반면 순차 패턴을 발견하는 것은 트랜잭션 상호간의 문제인 것이다. 각 고객들의 트랜잭션을 시간 순서로 볼 수 있는데 이것을 소비자 순차집합(customer sequence)이라고 한다. 소비자 순차집합의 형태는 <항목집합 (T_1) 항목집합 (T_2)…항목집합 (T_n)>의 시퀀스이다. 시퀀스가 특정 고객에 대한 소비자 순차집합에 속해 있다면 그 고객은 이 시퀀스를 지지한다고 말할 수 있다. 시퀀스에 대한 지지도의 정의는 시퀀스를 지지하는 전체 고객들의 수이다. 주어진 고객에 대한 트랜잭션 데이타베이스 D에서 순차 패턴 탐사는 사용자가 정의한 최소 지지도를 만족하는 모든 시퀀스들 사이에서 최대 시퀀스를 찾는 것이다. 이러한 각각의 최대 시퀀스들을 순차 패턴(sequential patterns)이라고 하고 최소 지지도를 만족하는 시퀀스를 빈발 시퀀스(large sequence)라고 부른다.

바코드의 발달로 판매 조직들은 방대한 양의 판매 데이타를 수집하고 저장할 수 있게끔 되었는데 이 판매 데이타의 레코드들은 트랜잭션의 시간과 그 트랜잭션에서 포함되어 있는 항목들로 구성된다. 또한 이 데이타 레코드에는 고객의 ID도 포함되어 있다. 이러한 데이타에서 순차 패턴을 탐사한다고 하는 것은 예를 들어 그것이 비디오 대여 정보를 가진 데이타베이스라고 해보면 여기서 고객들의 대여 패턴을 찾는 것이다. 패턴의 예를 들어보면 Star Wars를 대여한 고객이 Empire Strikes Back을 대여하고 Return of the Jedi의 순서로 비디오를 대여하는 패턴을 찾는 것이다.

한가지 주의 할 점은 Star wars→Empire Strikes Back Return of the Jedi 사이에 다른 비디오가 대여되어도 그 패턴은 지지가 된다. 이렇게 소비자의 구매 패턴을 찾는 것 외에도 의료 분야에서 순차 패턴을 이용한 응용을 보면 환자들에 대한 질병과 투약에 관한 데이타를 이용해 일정 지지도 이상을 갖는 특정 질병에 대한 투약 과정을 순차 패턴으로 표현할 수 있다면 해당 질병 치료에 많은 도움을 주게되고 의사는 더 좋은 진료와 치료를 할 수 있게 된다.

4.3 순회 패턴

서류(documents)와 객체들(objects)이 상호 접근을 허용하도록 서로 연결되어 있는 환경에서 제공하는 분산된 정보를 접근하는 패턴을 탐사하는 새로운 데이타 마이닝의 기법을 연구하고 있다. 그런 정보를 제공하는 환경으로는 world wide web과 on-line service로 CompuServe, America Online, Hitel 등이 있다. 사용자는 관심있는 정보를 찾을 때 제공되는 하이퍼링크와 같은 것을 통해서 한 객체에서 다른 객체로 이동한다. 이런 환경하에서 사용자의 접근 패턴(access patterns)을 이해하면 시스템 설계를 개선시킬 수도 있고 더 좋은 마켓팅 결정으로 이끌어 낼 수도 있다. 시스템 설계면에서는 크게 연관된 객체들 사이에는 효율적인 접근을 제공하고 그 페이지에 더 좋은 저작 설계를 제공할 수 있다. 마켓팅 결정면에서는 적절한 위치에 광고를 함으로써 더 좋은 소비자/사용자의 분류와 행위 특성 분석을 제공할 수도 있다. 그런 환경 하에서 사용자의 접근 패턴을 포착하는 것을 순회 패턴(traversal patterns) 탐사[1, 19]라 일컫는다.

순회 패턴을 찾는 과정은 다음과 같다. 사용자의 로그 파일에서 한 사용자의 접근 패턴의 예로서 그림 4에서와 같이 먼저 순회 노드를 정리하면 {A, B, C, D, C, B, E, F, G, F, H, A, W, X, W, Y}로 이루어졌다. 이 접근 패턴에서 순 방향 참조 집합은 {ABCD, ABEFG, ABEFH, AWX, AWY}로 된다. 각 사용자에 대한 순 방향 참조 집합을 구하고, 모든 사용자들에 대한 순 방향 참조 집합에서 순차 패턴에서와 같이 최소 지지도를 만족하는 빈발 참조 시퀀스를 발견한다. 마지막으로 빈발 참조 시퀀스에서 최대 참조 시퀀스[19]를 찾아내는 것이 순회 패턴 탐사를 하는 것이다.

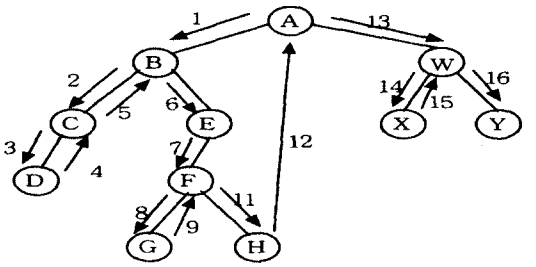


그림 4 접근 패턴(access patterns)의 예

4. 결 론

데이타 마이닝 분야 중에서 연관 규칙에 관한 정의와 연구 동향을 설명하였다. 연관 규칙 탐사는 응용성이 아주 높아 많은 연구가 이루어지고 있고, Data Warehousing을 구축할 적에 기본적인 도구로 활발히 이용되고 있다.

연관 규칙을 탐사하는 알고리즘의 연구의 초점은 빈발 항목집합을 효과적으로 찾아내기 위하여 후보 항목집합의 생성과 이 것을 저장하는 자료구조에 대한 연구가 많이 진행되었다. 그리고, 기본적인 연관 규칙을 응용한 순차 패턴, 순회 패턴, 주기적인 연관성 등에 연구가 이루어져 이를 활용하는 소프트웨어 개발에도 연구가 집중되고 있다. 연관 규칙을 활용하는 관점에서 사용자와 대화식으로 시스템이 구성되어 보다 유용한 규칙을 찾아내려는 시도가 이루어지고 있다. 이런 관점에서 탐사되어진 연관 규칙의 유용성에 관한 연구가 중요한 연구 토픽으로 떠오르고 있다. 연관 규칙의 중요도 및 관심도는 정의에 따라 그 규칙의 지지도와 신뢰도가 있고, 최근에는 확신도(conviction)과 개선도(improvement) 등이 유용성의 측정단위로 연구되고 있다. 앞으로 연관 규칙 탐사가 데이타 마이닝의 다른 분야에도 많은 영향을 끼치리라 생각된다.

참고문헌

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

[2] A. Berson and S.J. Smith, Data Warehousing, Data Mining, and OLAP, p. 612, McGraw-Hill, NewYork, 1997.

[3] R. Agrawal, T. Imielinski, and A. Swami, "Mining association rules in large databases", In *Proceedings of ACM SIGMOD Conference on Management of Data, Washington* D.C., pp. 207-216, May 1993.

[4] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules", In *Proceedings of the 20th VLDB Conference,* Santiago, Chile, Sept., 1994.

[5] H. Mannila, H. Toivonen, and A.I. Verkamo, "Efficient Algorithms for Discovering Association Rules", in *AAAI Workshop on Knowlege Discovery in Databases,* Eds. Usama M. Fayyad and Ramasamy Uthurusamy, pp. 181-192, Seattle, Washington, July 1994.

[6] A. Muller, "Fast sequential and parallel algorithms for association rulemining: a comparison", *University of Maryland-College Park CS Technical Report,* CS-TR-3515, 76 pages, August, 1995.

[7] M. Houtsma and A. Swami, "Set-Oriented mining for association rules", IBM Research Report, RJ 9567 (83573) October 22, 1993.

[8] J.S. Park, M.-S. Chen, and P.S. Yu, "An effective hash-based algorithm for mining association rules", In *Proceedings of ACM SIGMOD Conference on Management of Data,* pp. 175-186, San Jose, California, May, 1995.

[9] A. Savasere, E. Omiencinsky, and S. Navathe, "An efficient algorithm for mining association rules in large databases", In *Proceedings of the 21st VLDB Conference,* pp. 432-444, Zurich, Swizerland, 1995.

[10] H. Toivonen, "Sampling Large Databases for Association Rules", In *Proceedings of the 22nd VLDB Conference,* Mumbai(Bombay), India, 1996.

[11] J.S. Park, P.S. Yu, and M.-S. Chen, "Mining Association Rules with Adjustable Accuracy", In *Proceedings of ACM CIKM 97,* pp. 151-160, Las Vegas, Nevada, November, 1997.

[12] S. Brin, R. Motwani, J.D. Ullman, and S. Tsur, "Dynamic Itemset Counting and Implication Rules for Market Basket Data", In *Proceedings of ACM SIGMOD Conference on Management of Data,* Tucson, Arizona, pp. 255-264, May, 1997.

[13] B. Ozden, S. Ramaswamy, and A. Silberschatz, "Cyclic Association Rules", In *Proceedings of the 14th International Conference on Data Engineering,* pp. 412-421, Olrando, Florida, Feb, 1998.

[14] A. Savasere, E. Omiencinski, and S. Navathe, "Mining for Strong Negative Associations in a Large Database of Customer Transactions", In *Proceedings of the 14th International Conference on Data Engineering,* pp. 494-502, Olrando, Florida, Feb, 1998.

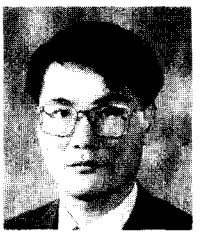
[15] R.T. Ng, L.V.S. Lakshmann, and J. Han, "Exploratory Mining and Pruning Optimizations of Constrained Association Rules", n *Proceedings of ACM SIGMOD Conference on Management of Data,* pp. 13-24, Seattle, Washington, June, 1998.

[16] R. Srikant and R. Agrawal, "Mining Generalized Association Rules", In *Proceedings of the 21st VLDB Conference, Zurich,* Swizerland, 1995.

[17] R. Agrawal and R. Srikant, "Mining sequential patterns", In *Proceedings of the 11th International Conference on Data Engineering,* Taipei, Taiwan, March, 1995.

[18] R. Srikant and R. Agrawal, "Mining Sequential Patterns: Generalizations and Performance Improvements", In *Proc. of the Fifth Int'l Conference on Extending Database Technology(EDBT),* Avignon, France, March 1996.

[19] M.-S. Chen. J.S. Park and P.S. Yu, "Data Mining for Path Traversal Patterns in a Web Environment", In *Proceedings of the 16th International Conference on Distributed Computing Systems,* pp. 385-392, Hong Kong, May, 1996.

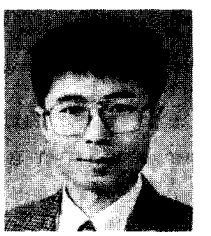


박 종 수

1981 부산대학교 전기기계공학과 학사
1983 KAIST 전기및전자공학과 석사
1983~1986 국방부 근무
1990 KAIST 전기및전자공학과 박사
1990~현재 성신여자대학교 전산학과 부교수
1994~1995 IBM Watson연구소 객원연구원
E-mail: jpark@cs.sungshin.ac.kr

유 원 경

1979 서울대학교 계산통계학 학사
1981 서울대학교 계산통계학 석사
1981~1986 한남대학교 전자계산학과 조교수
1986~현재 성신여자대학교 전산학과 교수
1987 서울대학교 계산통계학(전산학 전공) 박사
E-mail: wyoo@cs.sungshin.ac.kr



홍 기 형

1985 서울대학교 컴퓨터공학과 학사
1987 KAIST 전산학과 석사
1994 KAIST 전산학과 박사
1994~1998 ETRI DB분야 선임연구원
1998~현재 성신여자대학교 전산학과
E-mail: khhong@cs.sungshin.ac.kr