효과적인 빈발 항목 생성 알고리즘†

채 덕 진° 황 부 현
전남대학교 전산학과
{djchai, bhhwang}@sunny.chonnam.ac.kr

An Effective Large itemset Generation Algorithm

Duck-Jin Chai° Bu-Hyun Hwang
Dept. of Computer Science, Chonnam National University

요 약

대용량의 데이터베이스에서 여러 트랜잭션에 동시에 나타나는 항목들의 모임인 빈발 항목집합을 찾아내는 데이터 마이닝 방법을 연관 규칙 탐사라고 한다. 빈발 항목집합을 찾아내는 문제는 항목 집합들의 후보 집합을 생성하고 빈발 항목집합의 조건을 충족시키는 후보 집합을 추출함으로써 해결된다. 그리고 이러한 작업은 각각의 빈발 k-항목집합에 대해 k가 증가함에 따라 반복적으로 수행된다. 그러나 연관 규칙 탐사에 관한 기존의 연구는 주로 데이터베이스를 이루는 항목들의 수가 많거나 트랜잭션의 길이가 긴 경우의 대용량 데이터베이스에서 빈발 항목집합의 발견에 초점을 맞추고 있다. 본 논문에서는 데이터베이스를 이루는 전체 항목의 수가 적거나 트랜잭션의 크기가 작은 경우 효과적으로 빈발 항목집합을 찾을 수 있는 연관 규칙 탐사 방법을 제안한다. 그리고 성능 평가를 통하여 제안하는 방법의 성능 및 타당성을 보인다.

1. 서론

많은 분야에서 데이터베이스에 저장되는 데이터의 양이 증가하고, 데이터베이스의 응용 범위가 확대됨에 따라 대용량 데이터베이스로부터 유용한 지식을 발견하고자 하는 데이터 마이닝(data mining) 기술에 대한 연구가 활발히 진행되고 있다[5]. 데이터 마이닝 기술은 특성화(characterization), 분류화(classification), 군집화(clustering), 연관 규칙 탐사(association), 경향 분석(trend analysis), 패턴 분석(pattern analysis) 등으로 나눌 수 있다. 연관 규칙 탐사(mining association rules)는 가장 중요한 데이터 마이닝 방법들 중 하나이다[7].

연관 규칙 탐사는 트랜잭션에 나타나는 항목의 발생 빈도수가 최소 지지도(minimum support)보다 큰 빈발 항목집합(large itemsets)을 찾는 방법이다. 지금까지 빈발 항목집합을 찾기 위한 다양한 연관 규칙 탐사 알고리즘들이 제시되었다[1,2,3,4]. 제안된 기존의 연관 규칙 탐사 알고리즘들은 먼저 빈발 항목 집합들을 생성하기 위한 후보 집합을 만들고 그 후보 집합들 중에서 빈발 항목 집합들을 찾아낸다.

그리고 한번의 반복시행에서 발견된 빈발 항목 집합들은 다음 반복과정 에서 후보 집합을 생성하기 위해 사용되고, 이 과정은 더 이상의 후보 집합을 만들 수 없을 때까지 반복된다.

기존의 연관 규칙 탐사 알고리즘의 성능평가를 보면 트랜잭션의 길이나 최소 지지도의 변화에 따라 많은 성능상의 차이가 있음을 알 수 있다[3,6]. 각 트랜잭션은 다양한 형태로 이루어진다. 판매 트랜잭션에서 하나의 트랜잭션이 하루동안 판매되는 품목들이라고 가정하면, 트랜잭션은 판매되는 품목에 따라 다양한 길이를 가질 수 있다.

본 논문에서는 트랜잭션의 길이가 작거나 데이터베이스를 이루는 항목들의 수가 비교적 적은 트랜잭션 데이터베이스에서 최소 지지도에 영향을 받지 않고 기존의 알고리즘보다 더 짧은 시간에 빈발항목집합을 발견할 수 있는 효과적인 연관 규칙 탐사 알고리즘을 제안한다. 제안하는 방법은 해쉬 기법에 기초하여 한번의 데이터베이스 조사 과정을 통하여 빈발 k-항목집합들을 추출한다. 이에 대해서는 3장에서 자세히 설명할 것이다. 논문의 구성은 다음과 같다. 2장에서는 연관 규칙과 기존의 알고리즘에 대해 설명하고, 3장에서는 본 논문에서 제안하는 트랜잭션의 길이가 작은 데이터베이스에서의 연관 규칙 탐사 알고리즘을 기술한다. 그리고 4장에서는 모의 실험을 통한 성능평가를 기술하고 마지막으로 5장에서 결론을 내린다.

† 이 논문은 한국과학재단 1999년도 특정기초연구비 (1999-2-303-006-3) 지원에 의하여 연구되었음.

2. 관련 연구

2.1 연관 규칙

연관 규칙이란 특정 사건 집합의 발생이 다른 사건의 발생을 암시하는 경향을 표현하는 규칙으로 다음과 같이 정의할 수 있다. $I=\{i_1, i_2, \ldots, i_m\}$을 항목(item)이라 불리는 문자들의 집합이라 하자. 데이터베이스 D는 트랜잭션들의 집합이며, 하나의 트랜잭션 t는 항목의 집합과 트랜잭션 식별자 TID를 포함한다. X와 Y를 항목들의 집합이라 하자. $X \subseteq I$인 항목 집합에 대하여, $X \subseteq t$이면 X는 t에 포함되었다고 한다. 이때 연관 규칙은 $X \Rightarrow Y$로 표현되며, X를 규칙의 조건부(antecedent), Y를 결과부(consequent)라 한다. 이때 $X \subseteq I$, $Y \subseteq I$이고 $X \cap Y = \emptyset$이다[2].

2.2 DHP 알고리즘

[3]에서는 연관 규칙 탐사 알고리즘으로 DHP 알고리즘을 제안하였다. DHP 알고리즘은 해쉬 기법을 이용하여 빈발 항목집합을 효율적으로 생성하고 트랜잭션의 크기를 효과적으로 줄일 수 있다. DHP에 의해 생성된 후보 항목집합의 수는 양적으로 기존 방법에 비해 더 적기 때문에 전체 프로세스의 수행상의 병목을 개선한다. 덧붙여 DHP는 트랜잭션 데이터베이스 크기를 점차적으로 감소시키기 위해 효과적인 전지 기법을 사용한다. DHP는 다음 후보 항목집합의 생성을 위해 불필요한 항목집합들을 삭제하는 해쉬 기법을 사용한다. 후보 k-항목집합들의 지지도를 데이터베이스의 스캔으로 계산할 때, 전지후의 각 트랜잭션의 모든 가능한 (k+1)-항목집합들을 해쉬 테이블 H_{k+1}에 해쉬하는 방법으로 DHP는 미리 후보 (k+1)-항목집합들의 관한 정보를 축적한다. 이 해쉬 테이블에 속한 각 버켓은 이제까지 이 버켓에 해쉬 되어진 항목집합들의 개수를 나타내는 수를 포함한다.

DHP는 2-후보 항목 집합을 생성하기 위해서 미리 데이터베이스를 스캔하여 생성된 해쉬 테이블을 참조하기 때문에 최소지지도가 높은 경우에는 기존의 알고리즘보다 낮은 성능을 보일 수 있다[6].

3. DLG(Direct Large itemsets Generation) 알고리즘

연관 규칙 탐사는 각 트랜잭션에서 동시 발생하는 항목들의 연관 관계를 추출하는 것이다. A⇒B와 같은 형태의 연관 규칙을 추출하기 위해 기존의 알고리즘들이 서로 매우 다름에도 불구하고 그들 모두는 최소 지지도를 만족하는 빈발 항목집합을 추출하고 빈발 항목집합에서 최소 신뢰도를 만족하는 규칙을 추출하는 두 단계의 기본적인 스키마를 사용한다. 본 논문에서는 하나의 트랜잭션이 하루에 판매되는 품목들로 이루어진다고 정의 할 때, 많은 종류의 생필품과는 상대적으로 적은 종류의 품목들에 대해서 효과적으로 수행할 수 있는 DLG 알고리즘을 소개한다. 연관 규칙은 하나의 트랜잭션을 이루는 항목들의 집합 안에 존재한다. 예를 들어, A⇒B라는 연관 규칙이 존재할 때 A와 B는 하나의 트랜잭션 안에 존재한다. 그러므로, 트랜잭션에서 만들어질 수 있는 모든 부분집합들을 생성하면 생성된 부분집합들 중 하나는 하나의 연관 규칙이 될 수 있다는 말로 바꿀 수 있다. DLG 알고리즘은 기본적으로 이러한 성질을 이용한다.

알고리즘은 크게 두 부분으로 이루어진다. 첫 번째 부분에서는 트랜잭션에서 발생 가능한 k-항목집합들을 해쉬함수를 통해 각 버켓에 카운트한다. 두 번째는 각 버켓에 존재하는 값과 최소 지지도를 비교하여 지지도 이상인 버켓들을 추출하면 이들이 빈발 항목집합이 된다.

TID	2 - Itemsets	3 - Itemsets	4 - Itemsets
100	AC, AD, CD	ACD	
200	BC, BE, CE	BCE	
300	AB, AC, AE, BC, BE, CE	ABC, ABE, ACE, BCE	ABCE
400	BE		

<그림 1> 트랜잭션 분류

그림 1은 트랜잭션을 k-부분집합별로 분류하는 예를 보여준다. 먼저 TID가 100인 트랜잭션 A, C, D를 읽는다. 트랜잭션의 길이가 3이므로 2-부분집합과 3-부분집합이 생성될 수 있다. 각 부분집합을 생성하는 것은 기존의 알고리즘에서 사용하던 방법을 적용한다. 즉, A, C, D를 조인하여 2-부분집합인 AC, AD, CD가 생성되고 3-부분집합인 ACD가 생성된다. 연관 규칙 탐사에서 1-부분집합은 기존의 알고리즘에서 보듯이 다음단계의 후보를 생성하기 위해서는 필요하지만, 항목들의 연관성을 찾는 탐사에서 1-빈발 항목집합은 의미가 없다.

AC	AD	CD	ACD	k-Itemsets	
1	1	1	1	Counted value	100(TID) 수행결과
13	14	34	134	Hash address	

AC	AD	BC	BE	CD	CE	ACD	BCE	
1	1	1	1	1	1	1	1	200(TID) 수행결과
13	14	23	25	34	35	134	235	

AB	AC	AD	AE	BC	BE	CD	CE	ABC	ACD	ACE	BCE	ABCE	
1	2	1	1	2	2	1	2	1	1	1	2	1	300(TID) 수행결과
12	13	14	15	23	25	34	35	123	134	135	134	1235	

AB	AC	AD	AE	BC	BE	CD	CE	ABC	ACD	ACE	BCE	ABCE	
1	2	1	1	2	3	1	2	1	1	1	2	1	400(TID) 수행결과
12	13	14	15	23	25	34	35	123	134	135	134	1235	

h{{x y}}=(order of x)*10+(order of y)
h{{x y z}}=(order of x)*100+(order of y)*10+(order of z)
h{{w x y z }}=(order of w)*1000+(order of x)*100+(order of y)*10+(order of z)

<그림 2> 알고리즘 수행 과정

그림 2는 각 트랜잭션을 분류하고 분류된 부분집합들이 해쉬함수를 통하여 해당하는 버켓의 값을 1씩 증가시키는 예를 보여준다. 먼저, TID가 100인 트랜잭션을 분류하고 분류된 AC, AD, CD, ACD는 해당하는 버켓 13, 14, 34, 134번지의 값에 1씩 더해진다. 200인 트랜잭션을 분류한 후, 해당하는 버켓이 존재하면 그 버켓의 값에 1을 증가시키고 해당하는 버켓이 없으면 새로운 버켓을 생성하고 초기값을 1로 설정한다. 같은 방법을 다른 트랜잭션에 적용하고 마지막 트랜잭션까지 수행을 끝내면 TID가 400인 트랜잭션을 수행한 결과와 같은 표

를 얻을 수 있다.

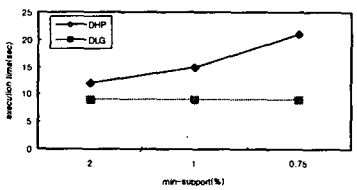
k-빈발 항목집합은 최소 지지도 이상인 값을 가지고 있는 버켓을 찾음으로써 해결된다. 그림 4에서 최소 지지도를 2라고 했을 때 2-빈발 항목집합은 AC, BC, BE, CE를 얻을 수 있고, 3-빈발 항목집합은 BCE를 얻을 수 있다. 모든 버켓을 검사하여 k-빈발 항목집합을 찾으면 알고리즘은 종료된다. 그림 3은 DLG 알고리즘을 기술한 것이다.

```
Forall   transactions t        ∈   D do begin
    Scan t to Database;
    for(k=2; k < |t|; k++) do begin
        k - itemsets      classification;
        increase 1 to the number of items hashed to bucket;
    end
end
for(b=1; b < |bucket|; b++) do begin
        Large    itemset     = { b.value >          minsup    };
end
```

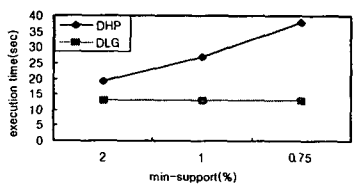
<그림 3> 알고리즘

4. 성능 비교

그림 4는 기존의 DHP 알고리즘과 본 논문에서 제안하는 DLG 알고리즘의 수행 성능을 보여준다. 주어진 데이터의 트랜잭션들의 개수가 10,000일 때 최소 지지도를 0.5%, 1.0%, 2.0%로 변화시키면서 측정한 결과이다. 결과에서 보듯이 DLG는 한번의 데이터베이스 스캔과 해쉬 기법을 사용하기 때문에 지지도에 크게 영향을 받지 않는다. 최소 지지도가 높거나 각 트랜잭션을 이루는 항목집합의 수가 많다면 DHP의 수행시간과 비슷해지거나 더 늘어날 수 있다. 그러나 일반적으로 수행 성능 상에서 우수함을 볼 수 있다. 그러므로, 트랜잭션 데이터베이스의 특성에 따라서 적절한 알고리즘의 선택이 중요하다고 할 수 있다.



(a) T5I4D10



(b) T10I4D10

<그림 4> DHP와 DLG의 성능 비교

5. 결론

본 논문에서 판매 트랜잭션의 방대한 데이터베이스에 있는 항목들간의 연관 규칙 탐사문제를 다루었다. 빈발 항목집합들을 발견하는 문제는 먼저 항목집합들의 후보 집합을 구성하고 이 후보 집합 내에서 빈발 항목집합들의 요구 조건을 만족시키는 항목집합들을 규명함으로써 해결할 수 있다. 본 논문에서는 판매 트랜잭션 데이터베이스를 이루는 각 트랜잭션의 길이가 비교적 짧은 경우와 전체 데이터베이스를 이루는 항목의 수가 적은 경우에 해쉬 기법을 사용하여 효과적으로 빈발 항목집합을 생성하는 알고리즘을 제안하였고, 모의 실험을 통하여 기존의 알고리즘을 적용했을 때 보다 성능이 더 좋다는 것을 증명하였다.

6. 참고 문헌

[1] R. Agrawal, T. Imielinski, and A. Swami, "Mining Association Rules between Sets of Items in Large Databases", *Proceedings of ACM SIGMOD*, pages 207-216, May 1993.

[2] R. Agrawal and R. Srikant, "Fast algorithms for mining Association Rules in Large Databases", In *Proceedings of the 20th VLDB Conference*, Santiago, Chile, Sept., 1994.

[3] J.S. Park, M.-S. Chen, and P.S. Yu, "An Effective Hash-Based Algorithm for Mining Association Rules", *Proceedings of ACM SIGMOD*, pages 175-186, May 1995.

[4] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and A.I. Verkamo, "Fast Discovery of Association Rules", In *Advances in Knowledge Discovery and Data Mining* edited by U.M. Fayyad, et al., pages 307-328, AAAI Press/The MIT Press, Menlo Park, 1996.

[5] Peiter Adriaans and Dolf Zantinge, "Data Mining" Addition Wesley Longman.

[6] 박종수, "대용량 데이터베이스상의 효과적인 관련 규칙 탐사를 위한 데이터 전지기법", 한국 정보과학회 데이터베이스 연구회지 제12권 제 4호, pages 59-75, 1996.

[7] 김정자, 이도헌, "데이터 마이닝 기술 및 연구동향", 한국 정보과학회 정보과학회지 제 16권 제 9호, pages 6-14, 1998.