공간 압축 및 효율적 탐사 기법을 이용한 빈발 폐쇄 항목집합 마이닝

박귀정° 한영우 이수원
숭실대학교 컴퓨터학과
redox@ibizup.co.kr°, ywhan@ksd.or.kr, swlee@computing.ssu.ac.kr

Frequent Closed Itemset Mining by Using a Space Compression and Efficient Search Technique

Gui-Jung Park° Young-Woo Han Soo-Won Lee
Dept. of Computing, Soongsil University

요 약

연관 규칙 마이닝은 일반적으로 많은 빈발항목집합과 연관 규칙을 생성하며, 생성된 연관 규칙은 상호 포함관계에 있거나 중복되는 경우가 많다. 이는 효과적인 마이닝 뿐 아니라 마이닝의 활용 효용성을 떨어뜨린다. 이를 해결하기 위하여 연관 규칙 마이닝과 동일한 성능을 가지며 생성되는 규칙의 수를 줄일 수 있는 빈발 폐쇄 항목집합 마이닝이 제안되었다. 본 연구에서는 연관규칙 마이닝 방법 중 가장 우수한 성능을 가지는 ARCS 알고리즘을 개선한 빈발 폐쇄 항목집합 마이닝을 제안한다.

1. 서 론

연관 규칙 마이닝은 두 단계로 구분되어 지는데 최소 지지도 임계치 이상의 지지도를 갖는 항목집합인 빈발항목집합(Frequent itemset)을 찾아내는 단계와 그것들로부터 연관 규칙을 생성하는 단계로 나뉜다. 첫 번째 단계인 빈발 항목집합 생성은 연관 규칙 마이닝 뿐만 아니라 순차 패턴(sequential pattern), 다차원패턴(multi-dimensional pattern), 최대 패턴(max-pattern) 마이닝의 기반이 되는 역할을 하고 있다. 두 번째 단계에서는 생성된 빈발항목집합들 중 최소신뢰도 임계치 이상인 규칙들을 찾아낸다.

연관 규칙 마이닝 알고리즘들은 종종 다량의 중복되는 빈발항목집합들을 유도하기도 하며, 중복되는 연관 규칙을 유도하고 있다. 이러한 중복을 방지하기 위한 최근의 연구들은 연관 규칙 마이닝과 동일한 효과를 얻을 수 있으면서 생성되어지는 규칙의 수를 줄일 수 있는 빈발 폐쇄 항목집합(frequent closed itemset)과 이에 대응하는 규칙을 생성하는 연구이다. 중복되는 규칙의 수를 줄일 수 있는 것은 생성되는 빈발항목집합의 수를 줄임으로써 가능하다.

예를 들어, 두 개의 트랜잭션으로 구성된 데이터베이스 $\{(a_1, a_2, \ldots, a_{100}),(a_1, a_2, \ldots,a_{50})\}$ 에서 최소 지지도의 임계치는 1이고(즉, 모든 항목집합이 빈발한 것이다), 그리고 최소 신뢰도 임계치가 50%일 때, 기존의 연관 규칙 마이닝 방법들은 (a_1), ..., (a_{100}), ..., (a_{99},a_{100}), ..., $(a_1, a_2, \ldots, a_{100})$와 같은 $2^{100}-1$(약 10^{30})정도의 빈발 항목 집합과 엄청난 수의 연관 규칙들을 생성한다. 그러나 빈발 폐쇄 항목집합 마이닝은 $\{(a_1, a_2, \ldots,a_{50}),(a_1, a_2, \ldots, a_{100})\}$와 같은 단 두 개의 빈발 폐쇄 항목집합들과 "$(a_1, a_2, \ldots,a_{50}) \Rightarrow (a_{51}, a_{52}, \ldots, a_{100})$"라는 단 하나의 연관 규칙만을 생성하고 다른 규칙들은 이 규칙으로부터 모두 유도 하는 방법이다[1].

본 논문은 빈발항목집합 마이닝 알고리즘인 ARCS (Association Rule mining in Compressed Space)[2] 알고리즘을 개선한 빈발 폐쇄 항목집합 마이닝 방법을 제안한다. 2장에서는 본 연구에 대한 관련 연구를 소개하고, 3장에서는 ARCS를 이용한 빈발 폐쇄 항목집합 마이닝을 설명하며, 4장에서는 실험결과에 대하여 소개한다.

2. 관련연구

2.1 연관규칙

항목들의 집합 $I=\{i_1, i_2, \ldots,i_m\}$이 주어지면, 트랜잭션 T는 I의 부분집합으로 정의된다($T \subseteq I$). 항목집합 $A \subset I$의 지지도를 Support(A), 최소지지도를 S_{min}이라 하면, $Support(A) \geq S_{min}$을 만족할 때 A는 빈발항목집합이 된다. 연관 규칙은 빈발항목집합으로 표현된 트랜잭션에서 각 항목간의 연관성을 반영하는 규칙이고, $R: X \rightarrow Y$로 나타낸다. 이때 규칙은 $X,Y \subset I$, $X \cap Y = \emptyset$의 특성을 갖는다. 연관 규칙의 타당성을 검증하기 위한 척도로서 지지도(support)와 신뢰도(confidence)가 사용되며 다음과 같이 표현된다.

$$Support(A, B) = P(A \cap B) \quad 식(1)$$

$$Confidence(A, B) = P(B|A) = P(A \cap B)/P(A) \quad 식(2)$$

그 외에 흥미도(interest), 확신도(conviction)등의 척도가 지지도와 신뢰도를 보완하기 위해 사용된다.

2.2 빈발 폐쇄 항목집합

빈발 폐쇄 항목집합 마이닝은 연관 규칙 마이닝과 동일하게 2단계로 이루어진다. 첫 번째 단계는 최소지지도 임계치 이상을 갖는 빈발 폐쇄 항목집합들을 생성한다.

두 번째 단계는 빈발 폐쇄 항목집합으로부터 최소신뢰도 이상의 연관규칙을 생성한다.

빈발 폐쇄 항목집합에 대한 정의는 다음과 같다. 항목집합 X와 동일한 지지도를 가지며 X를 포함하는 항목집합 X'가 존재하지 않으면 X를 폐쇄 항목집합이라 하고, 폐쇄 항목집합 X의 지지도가 최소지지도 임계치 이상이면 폐쇄 항목집합 X는 빈발하다고 한다. 즉 항목집합 X는 동일한 지지도를 갖는 항목집합들 내에서 최대항목집합(Maximal Itemset)이어야 한다. 연관 규칙의 생성은 일반적인 연관 규칙 생성과 동일하게 빈발 폐쇄 항목집합 X가 있고, X의 부분집합 Y(Y⊂X,Y≠∅)가 있을 때, Y⇒X-Y인 규칙이 생성된다. 최소지지도가 20이고, 최소신뢰도가 50%일 때, 트랜잭션데이터와 최소지지도 미만의 항목들을 제거한 여과 트랜잭션데이터를 나타내고 있는 <표1>를 통해 빈발 폐쇄 항목집합의 마이닝 결과 및 생성된 규칙은 다음과 같다.

<표1> 트랜잭션데이터베이스와 여과 트랜잭션 집합

TID	트랜잭션데이터	여과트랜잭션데이터
10	a,c,d,e,f	a,c,d,e,f
20	a,b,e	a,e
30	c,e,f	c,e,f
40	a,c,d,f	a,c,d,f
50	c,e,f	c,e,f

<표2> <표1>의 빈발항목집합 및 빈발폐쇄항목집합

지지도	빈발항목집합	빈발폐쇄항목집합
2	{a,c,d,f},{a,d,f},{a,c,f},{a,c,d},{c,d,f},{a,f},{d,f},{a,e},{a,d},{a,c},{c,d},{d}	{a, c, d, f},{a, e}
3	{c, e, f},{c, e},{e, f},{a}	{c, e, f},{a}
4	{c, f},{f},{e},{c}	{c, f},{e}

지지도 3인 빈발항목집합과 빈발폐쇄항목집합을 비교해 보면 빈발항목집합 {c,e},{e,f}는 {c,e,f}의 부분집합임으로 삭제된다. 다른 경우도 이와 동일하다.

연관 규칙 생성을 보면, 빈발 폐쇄 항목집합 {a,c,d,f}를 예를 들어보면 {c,f}는 {a,c,d,f}의 부분집합이고 지지도는 4이다. 이를 이용하여 {c,f}⇒{a,d}인 규칙은 신뢰도(2/4) 50%로 최소신뢰도를 만족한다. 빈발 폐쇄 항목집합으로부터 다른 규칙들을 유도해 보면, X⇒Y(신뢰도), {a}⇒{c,d,f}(67%), {e}⇒{c,f}(75%), {c,f}⇒{e}(75%), {e}⇒{a}(50%), {a}⇒{e}(67%)등의 규칙을 얻을 수 있다.

2.3 ARCS

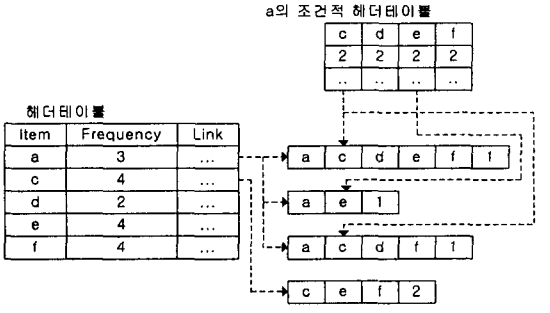
ARCS는 후보빈발항목집합을 생성하지 않는 비 Apriori계의 대표적 알고리즘인 FP-growth[3]와 H-mine[4]의 단점을 보완하여 만든 알고리즘이다. FP-growth는 빈발항목집합 발견시 반복적으로 조건적 FP-struct를 생성함으로써 공간적 시간적 낭비를 초래한다. 또한 대용량의 sparse한 데이터인 경우 FP-tree의 압축율이 낮아지고, 최소 지지도가 매우 낮을 경우 원본 트랜잭션 데이터베이스보다 FP-struct가 더 커질 경우가 발생할 수 있다.

H-Mine은 입력 데이터가 dense/sparse한지에 대한 명확한 기준 제시가 어려울뿐더러, dense한 경우 FP-struct를 사용함으로 FP-growth가 가지고 있는 단점을 내재하게되고, sparse한 경우 H-struct를 사용하므로 압축 효과를 기대하기가 어렵다. 또한 H-struct를 사용할 경우 사용하지 않는 링크공간으로 인한 저장공간의 낭비가 발생하고, 빈발항목집합 발견과정시 해당 항목에 대한 링크가 부족할 경우 backtracking을 해야 하는 문제점이 있다.

이러한 문제점을 해결하기 위해 CT(Compressed Transactions)-struct 자료구조와 CT-struct를 기반으로 하여 연관 규칙을 마이닝하는 방법이 ARCS 알고리즘이다. CT-struct는 동일한 항목집합으로 이루어진 여과된 트랜잭션(Filtered Transactions)에 대해 같은 공간을 사용하여 카운트 정보를 추가하고 Compressed Transaction을 항목이름만으로 구성하여 2차원 스트링 배열로 표현함으로써 효율적으로 저장공간 크기를 줄인다.

ARCS 알고리즘은 데이터 밀집도에 상관없이 전체 트랜잭션 데이터베이스의 여과된 트랜잭션 집합을 CT-struct로 압축한 후, 패턴-성장 방법을 사용하여 빈발항목집합을 발견한다. 또한 조건적 헤더테이블만을 반복적으로 생성하며, Backtracking 과정 없이 마이닝을 수행하므로 빠르게 빈발항목집합을 마이닝할 수 있다.

<표1>에 대한 ARCS의 CT-struct, 헤더 테이블 그리고 a의 조건적 헤더테이블은 <그림1>과 같다.



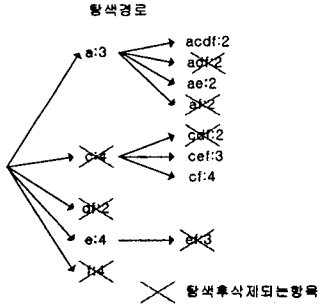
<그림1> <표1>의 CT-struct 와 헤더테이블

2개의 빈발 트랜잭션 {c,e,f}가 같은 저장 공간을 사용하고 있으며, 각 항목 사이에 불필요한 링크공간이 없다. 'a'에 대한 빈발항목 집합 마이닝을 하면서 두 번째 아이템인 'c'에 대한 링크를 헤더테이블에 추가하여 'a'에 대한 마이닝이 끝난 후, backtracking 작업 없이 'c'에 대한 완전한 링크 정보를 얻을 수 있고, 'a'의 링크 집합을 헤더테이블로부터 제거하여 임시 저장공간의 확대를 방지한다.

'a'의 조건적 헤더테이블을 생성한 후 최소지지도 미만의 항목들을 제거한 후 {(a,c):2} {(a,d):2} {(a,e):2} {(a,f):2}와 같이 4개의 빈발 항목을 찾을 수 있고, 다시 ac를 포함하는 조건적 헤더테이블을 생성한 후 빈발항목을 찾는 것을 반복적으로 수행한다.

3. ARCS를 이용한 빈발 폐쇄 항목집합 마이닝

기존의 빈발항목집합 마이닝 알고리즘들 중 하나인 ARCS는 모든 빈발항목집합들을 찾는 것이 목적이었지만, 이를 개선하여 빈발 폐쇄 항목집합들을 찾는 것은 동일한 지지도를 갖는 빈발항목집합들 내에서 최대항목집합(Max Itemset)을 찾는 것이다. ARCS는 조건적 헤더 테이블의 모든 항목들을 방문하여 빈발항목집합을 찾았으나, 이를 개선하여 조건적 헤더테이블의 모든 항목들의 지지도가 헤더테이블의 선택항목의 지지도와 동일할 경우 더 이상의 마이닝을 진행하지 않고 빈발 폐쇄 항목집합으로 결정한다. 이로 인해 탐색공간의 크기를 효과적으로 줄일 수 있다. <그림2>는 빈발 폐쇄 항목집합을 찾는 탐색공간을 보여주고 있다. <그림1>에서 보면 헤더 테이블에서 {a}의 지지도는 3이고 {a}의 조건적 테이블에 있는 항목들의 데이터는 모두 2로 서로 다르므로 {a:3}을 빈발 폐쇄 항목으로 생성한다. 이후 {ac}의 조건적 헤더 테이블을 생성하면 {d:2,f:2}와 같이 된다. {ac}의 지지도가 2이고 {ac}의 조건적 헤더테이블의 모든 항목들의 지지도가 2로 동일하므로 {acdf:2}가 빈발 폐쇄 항목집합으로 생성된다. 다시 {ad}의 조건적 헤더 테이블을 생성하면 {f:2}와 같이 된다. {ad}의 지지도가 2이고 조건적 헤더테이블 {f}의 지지도가 2로 같으므로 빈발 폐쇄 항목집합으로 선정되지만, 이미 빈발 폐쇄 항목집합인 {acdf}의 부분집합이므로 추가되지 않고 삭제된다.



<그림2> 빈발 폐쇄 항목집합 마이닝 탐색공간

4. 실험 및 결과

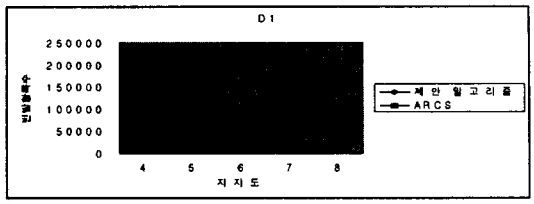
실험 데이터는 백화점 데이터와 영화 데이터를 사용하였으며, <표3>은 데이터의 특성을 나타낸다.

<표3> 실험데이터 특성

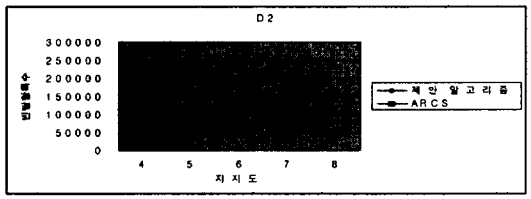
데이터	아이템 수	각 트랜잭션 당 평균아이템 수	전체 트랜잭션 수	사이즈	밀집도
영화 데이터(D1)	249개	36개	127개	56Kb	14.5
백화점 데이터(D2)	4017개	7개	26580개	4830Kb	0.17

실험에 사용된 알고리즘은 모두 JAVA로 구현하였고, 2GHz Pentium4, 512M, 40G hard disk로 구성된 PC에서 이루어졌다.

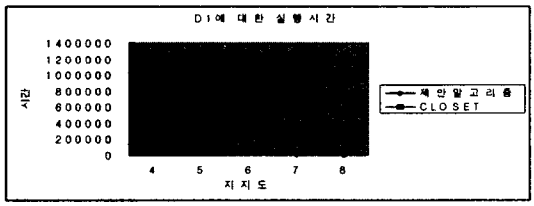
<그림3>과 <그림4>는 각각 D1과 D2에 대한 지지도 대비 빈발항목집합 수를 보여주고 있다. <그림5>와 <그림 6>은 FP-tree를 이용한 빈발 폐쇄 항목집합 알고리즘인 CLOSET[1]과 제안한 알고리즘의 실행시간을 비교한 것이다.



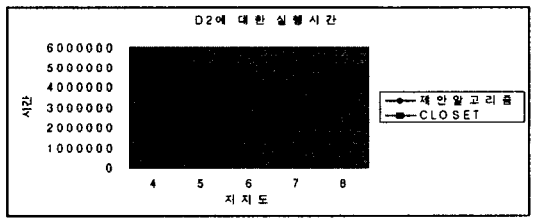
<그림3> D1에대한 지지도 대비 빈발항목집합 수



<그림4> D2에 대한 지지도 대비 빈발항목집합 수



<그림5> D1에 대한 제안알고리즘과 CLOSET의 실행시간



<그림6> D2에 대한 제안알고리즘과 CLOSET의 실행시간

5. 결론

본 논문에서는 일반적인 연관규칙 마이닝에서 과도한 빈발항목집합 생성으로 인해 불필요한 연관 규칙들이 유도되는 점을 개선하기 위하여 빈발 폐쇄 항목집합을 마이닝하는 방법을 ARCS 알고리즘을 이용하여 제안하였다. 실험결과 제안방법이 ARCS에 비하여 생성된 연관규칙의 수를 줄일 수 있었으며, CLOSET에 비하여 실행시간을 단축할 수 있었다.

6. 참고문헌

[1] J. Pei, J. Han, and R. Mao. "CLOSET: An efficient algorithm for mining frequent closed itemsets". In DMKD 2000, pp. 11-20

[2] 한영우, 2002, "고속의 연관규칙 마이닝을 위한 효율적 공간 압축 및 탐사 기법", 숭실대학교 석사학위 논문

[3] Han, J., Pei, J., Yin, Y., "Mining Frequent Patterns without Candidate Generation". SIGMOD'00 (2000) 1-12

[4] Pei, J., Han, J., Lu, H., Nishio, S., Tang, S., Yang, D., "H-mine: Hyper-Structure Mining of Frequent Patterns in Large Databases". ICDM (2001) 441-448