한국지능시스템학회 논문지 2009, Vol. 19, No. 2, pp. 285-290

T-알고리즘을 이용한 연관규칙의 효과적인 감축

An Effective Reduction of Association Rules using a T-Algorithm

박진희 · 정환묵
Jin-Hee Park and Hwan-Mook Chung

대구가톨릭대학교 컴퓨터정보통신공학부

요 약

데이터에 숨겨진 패턴을 탐색하는 데이터마이닝에서 가장 많은 연구가 이루어진 분야가 연관규칙 마이닝이다. 연관규칙 마이닝에서는 방대한 수의 트랜잭션 데이터를 다루게 되므로 고속처리 방식의 실현이 중요한 과제가 되고 있다. 그리고 연관규칙 탐사기법에서 규칙을 도출하는데 소요되는 시간은 데이터에 포함되어 있는 항목의 수에 비례하여 기하급수적으로 늘어나기 때문에 규칙의 수를 줄이는 과정이 필연적으로 요구된다.
본 논문에서는 트랜잭션 데이터 항목들을 이진형식으로 비교하여 연관성 규칙의 수를 효과적으로 감축할 수 있고 항목간의 지지도와 신뢰도를 함께 향상 시킬 수 있는 T-알고리즘을 제안하고 시뮬레이션을 통하여 확인하였다.

키워드 : 데이터마이닝, 연관규칙, 규칙감축, Apriori algorithm.

Abstract

An association rule mining has been studied to find hidden data pattern in data mining. A realization of fast processing method have became a big issue because it treated a great number of transaction data. The time which is derived by association rule finding method geometrically increase according to a number of item included data. Accordingly, the process to reduce the number of rules is necessarily needed.
We propose the T-algorithm that is efficient rule reduction algorithm. The T-algorithm can reduce effectively the number of association rules. Because that the T-algorithm compares transaction data item with binary format. And improves a support and a confidence between items. The performance of the proposed T-algorithm is evaluated from a simulation.

Key Words : Data mining, Association Rule, Rule reduction, Apriori algorithm

1. 서 론

지난 수십 년간 여러 가지 형태로 저장되어 있는 데이터의 양은 기하급수적으로 증가되어 왔다. 그러나 이러한 데이터의 무제한적인 증가는 우리가 원하는 정보를 찾아내는 일을 보다 어렵게 만들고 있다. 왜냐하면 우리는 대용량의 데이터로부터 의미 있는 지식(knowledge)을 찾아내고자 하는 것이 목적인데 반하여 실제적으로 오히려 데이터만 계속 쌓이고 있는 상황이기 때문이다. 이러한 상황에서 대용량의 데이터로부터 의미 있는 지식을 찾아내는 데이터마이닝(data mining)은 현재 중요한 문제 중 하나가 되었다[1].

데이터 마이닝과 관련된 기법으로는 이웃한 K-근사방법(K-nearest Neighbor Method), 의사결정트리, 연관규칙(association rule), 신경망이론과 유전자 알고리즘 등이 있다. 연관성 분석은 데이터의 자료구조가 간단하고 결과가 분명하여 검색에서 많이 사용되어 왔으나 항목수의 증가에 따라 계산양이 폭증하고 거래가 드문 품목에 대해서는 정보를 찾기가 어려운 단점이 있다.

접수일자 : 2008년 11월 1일
완료일자 : 2009년 2월 10일

따라서 본 논문에서는 데이터베이스에 저장되어 있는 방대한 데이터로부터 유용한 정보 및 지식을 추출하는 연관규칙을 기반으로 하여 기존의 연관규칙 기법의 단점을 보완한 알고리즘을 제안한다.

2. 관련 연구

2.1 연관규칙 마이닝

데이터에 숨겨진 패턴을 탐색하는 데이터마이닝에서 가장 많은 연구가 이루어진 분야가 연관규칙 마이닝이다[2]. 연관규칙은 대용량의 데이터베이스에서 어떤 사건들이 함께 발생하거나 또는 하나의 사건이 다른 사건을 암시하는 것과 같은 사건간의 상호관계를 마이닝하는 것으로 항목 X와 항목 Y사이의 X→Y 형태의 규칙을 찾아낸다. 여기서 항목은 일반적으로 이진(boolean)속성을 가지며 항목 X가 나타나면 항목Y도 나타날 가능성이 높다는 연관관계를 나타낸다.

예를 들면 '목요일, 기저귀→ 맥주'라는 규칙은 목요일 날 기저귀를 사는 고객은 맥주도 함께 구입한다. 라는 규칙을 나타낸다. 그리고 식(1),(2)에서 S를 연관규칙의 지지도

(Support), C를 신뢰도(Confidence)라고 정의한다. 단, N은 데이터베이스의 모든 트랜잭션 수를 나타낸다.

$$S = \frac{\#T[X \cup Y]}{N} = P(X \wedge Y) \quad (1)$$

$$C = \frac{\#T[X \cup Y]}{\#T[X]} = P(Y \mid X) \quad (2)$$

지지도는 식(1)과 같이 나타내며 데이터베이스 중에 X와 Y가 동시에 출현하는 확률이다. 신뢰도는 식(2)와 같이 나타낼 수 있고 X에서 Y가 발생할 조건부 확률이다. 즉, $0 \leq s \leq 1$ 및 $0 \leq c \leq 1$이 성립된다. X→Y라는 연관 규칙은 조건부 X가 성립했을 때 후건부 Y가 성립할 확률(조건부 확률)이 신뢰도로 주어진다. 지지도는 데이터베이스 중의 어느 정도의 트랜잭션에 대해 이 규칙을 적용할 수 있는지의 비율을 나타내고 있고 규칙의 범용성의 지표가 된다.

2.2 빈발 항목집합 생성

빈발항목집합을 찾기 위한 맹목적 접근법은 격자구조에서 각 후보 항목집합(Candidate itemset)에 대한 지지도 카운트를 결정하는 것이다. 여기서 지지도 카운트란 특정항목집합을 포함하는 트랜잭션의 개수이다. 이를 행하기 위해서 모든 트랜잭션들과 비교해야할 필요가 있고, 이것은 그림 1에서 보여주는 연산이다.

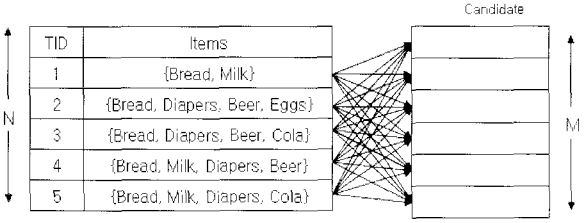


그림 1. 후보항목들의 지지도를 계산
Fig 1. Computation supportiveness of candidate itemset

만약 그 후보가 한 트랜잭션에 포함되어 있다면 그 지지도 카운트는 증가 할 것이다.

예를 들면 {Bread, Milk}에 대한 지지도는 그 항목 집합이 트랜잭션 1,4,5번에 포함되어 있기 때문에 세 번 증가한다. 이러한 접근법은 $O(NM\omega)$비교를 요구 하기 때문에 아주 비용이 많이 들 수 있다. 여기서 N은 트랜잭션의 개수, $M=2k-1$은 후보항목집합의 개수, 그리고 ω는 최대 트랜잭션 폭이다.

빈발 항목집합 생성의 계산 복잡도를 줄이는 데는 여러 가지 방법이 있지만 다음 두가지 방법이 대표적 방법이다.

1. 후보 항목 집합의 개수를 줄인다.
   Apriori 원리는 후보 항목 집합들 중에 일부의 지지도 값을 세지 않고 그것들을 제거하는 효과적인 방법이다.
2. 비교 횟수를 줄인다.
   모든 트랜잭션과 대조하여 각 후보 항목 집합을 맞추어 보는 대신, 후보 항목 집합들을 저장하거나 또는 데이터 집합을 압축하는 좀 더 고급의 자료 구조를 사용함으로써 비교 횟수를 줄일 수 있다.

2.2.1 Apriori알고리즘에서 빈발항목 집합생성

현실에서의 연관규칙 마이닝에서는 막대한 수의 트랜잭션 데이터를 다루기 때문에 고속처리방식의 실현이 중요하며 그중 Apriori 알고리즘이 가장 많이 사용되고 있다.

Apriori 알고리즘은 강한 연관성을 갖는 항목들을 발견하고 각 패스에 빈발 항목들을 발견하는데 초점을 둔다. 각 패스에 빈발항목들의 후보 항목집합을 구성하고 난 후 각 후보 항목집합의 발생 빈도수를 계산하고 사용자가 정의한 최소지지도의 최소 신뢰도를 기초로 하여 빈발 항목 집합들을 정의한다. 또한 Apriori 알고리즘의 문제점은 많은 항목들과 트랜잭션들이 알고리즘의 후반부 패스들에서 더 이상 필요 없음에도 불구하고 항상 각 패스마다 전체 데이터 셋을 검색해야 한다. 비 빈발 항목이나 트랜잭션들을 제거하는 것은 후보가 될 가능성이 없는 집합들을 카운트하기 때문에 빈발 항목 측정에 시간과 비용이 많이 드는 단점이 있다[4].

- Apriori 알고리즘의 연관규칙 탐색 단계[3]
  ① 트랜잭션데이터베이스 전체를 탐색하고 모든 사이즈 빈출 아이템 집합 전체 F를 구한다.
  ② 위의 스텝에서 구한 F를 탐색하고 주어진 최소 신뢰도 $S_{\min}$ 이상의 모든 상관규칙을 생성한다.

Apriori는 후보항목집합들의 기하급수적인 성장을 체계적으로 제어하기 위해 지지도 기반 가지치기의 사용법을 개척한 최초의 연관규칙탐사 알고리즘이다.

간단한 장바구니 트랜잭션의 예로 표 1과 같이 5개의 TID로 나타내보았다.

표 1. 장바구니 트랜잭션의 예제
Table 1. Example of basket transaction

TID	Items
1	{Bread, Milk}
2	{Bread, Diapers, Beer, Eggs}
3	{Bread, Diapers, Beer, Cola}
4	{Bread, Milk, Diapers, Beer}
5	{Bread, Milk, Diapers, Cola}

그림 2는 표 1에서 나타난 트랜잭션들에 대하여 Apriori 알고리즘의 빈발 항목 집합 생성 부분에 대한 설명을 보여준다.

최소 지지도 카운트 3과 같은 지지도 임계값을 60%로 가정한다. 처음에 각 후보 1-항목 집합으로 간주한다. 항목의 지지도를 계산한 후에 후보 항목 집합들{cola}와 {eggs}는 세 개의 트랜잭션 보다 더 작은 곳에서 나타나기 때문에 그 항목집합들은 버려진다. 다음 반복에서 Apriori 원리가 빈발하지 않은 것을 보증하기 때문에 후보 2-항목 집합들은 오직 빈발 1-항목 집합들만 사용하여 생성한다. 단지 4개의 빈발한1-항목 집합들이 있기 때문에 그 알고리즘에 의해 생성된 후보 2-항목 집합의 개수는 $\binom{4}{2} = 6$이다. 항목의 지지도값을 계산한 다음 6개의 후보중에서 항목 집합들 {Beer,Bread}와 {Beer, Milk}는 결과적으로 빈발하지 않은 것으로 알게 된다. 나머지 4개의 후보들은 빈발하고 후보 3-항목 집합들을 생성하는데 사용 될 것이다.

Apriori원리로 오직 부분집합이 빈발 후보 3-항목 집합들만 유지할 필요가 있다. 이런 성질을 만족하는 유일한 후보는 {Bread, Diapers, Milk}이다.

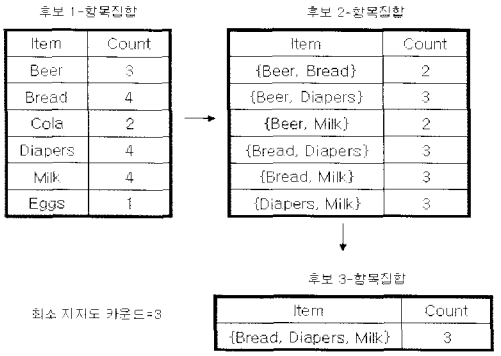


그림 2. Apriori 알고리즘을 사용한 빈발 항목 집합생성
Fig 2. Generation frequence itemset using apriori algorithm

Apriori 가지치기 전략의 유효성은 생성된 후보 항목 집합의 개수를 세어봄으로써 입증할 수 있다. 모든 항목 집합들 (크기 3까지)을 후보들로 열거하는 맹목적 전략은 $\binom{6}{1}+\binom{6}{2}+\binom{6}{3}=6+15+20=41$ 개의 후보를 생기게 할수 있을 것이다. Apriori 원리로 이 숫자는 $\binom{6}{1}+\binom{4}{2}+1=6+6+1=13$개의 후보로 감소하고 후보 항목 집합 개수에서 68% 감소율을 나타내었다.

다음은 Apriori 알고리즘의 빈발 항목 집합 생성 과정을 14단계로 나타내었다.

1 : k=1
2 : $F_k=\{i \mid i \in I \wedge \sigma(\{i\}) \geq N \times minsup\}$
//모든 1-집합탐색항목 탐색
3 : repeat
4 : k=k+1
5 : $C_k = apriori-gen(F_{k-1})$ //후보항목집합생성
6 : for each transaction $t \in T$ do
7 : $C_t = subset(C_k, t)$ //t에속하는 모든 후보들 식별
8 : for each candidate item set $c \in C_k$ do
9 : $\sigma(c)=\sigma(c)+1$ //지지도 카운트 증가
10: end for
11: end for
12: $F_k=\{c \mid c \in C_k \wedge \sigma(\{c\}) \geq N \times minsup\}$
//빈발 k-항목집합 추출
13: until $F_k=0$
14: Result = $\cup F_k$

C_k를 후보 k-항목들의 집합들의 집합이라하고 F_k를 빈발 k-항목들의 집합이라 정의 한다.

1번에서 3번 라인의 알고리즘은 각 항목의 지지도를 결정하기 위하여 전체 데이터 집합에 단일 패스를 시행한다. 이 단계를 마치면 모든 빈발 1-항목 집합들의 집합 F_1이 찾아 질 것이다.

4번에서 5번라인의 알고리즘은 이전 반복에서 찾아진 빈발(k-1)-항목집합들을 사용하여 새로운 후보 k-항목 집합들을 반복적으로 생성할 것이다.

후보 생성은 Apriori-gen이라 불리는 함수를 사용하여 구현된다.

6번에서 10번 라인의 후보들은 지지도를 카운트하기 위해 알고리즘은 전체데이터 집합에 걸쳐 추가적인 패스를 시행할 필요가 있다. subset 함수는 C_k에 속한 후보 항목 집합들중에서 각 트랜잭션 t에 포함된 모든 항목 집합들을 결정하는데 사용된다.

12번 라인 항목 집합들의 지지도를 계산한 다음 알고리즘은minsup보다 작은 지지도 카운트를 갖는 모든 후보 항목 집합들을 제거한다.

13번과 14번 라인은 새로운 빈발 항목 집합이 생성되지 않을 때 (즉, F_k=0) 알고리즘을 끝낸다[5].

3. 연관규칙 감축 알고리즘

3.1 연관규칙의 감축

이진형식으로 비교하는 T-알고리즘을 이용한 규칙의 감축순서는 그림 3과 같다.

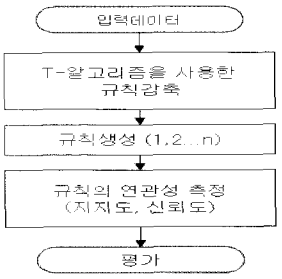


그림 3. T-알고리즘을 이용한 규칙감소
Fig 3. Reduction of rules using T-algorithm

입력된 트랜잭션데이터의 규칙을 T-알고리즘을 사용하여 규칙을 감축한다. 이렇게 감축된 n 개의 규칙은 규칙의 연관성을 측정하기 위하여 지지도와 신뢰도를 측정하게 된다.

트랜잭션 데이터 내에 존재하는 원 데이터를 본 논문에서 제안하는 T-알고리즘을 이용하여 Step 1에서 Step n까지 감축을 수행한다.

Step 4에서 1의 개수가 n개인 항과 n+1의 항을 비교를 해서 1이 차이가 나는 최소항 들을 골라내는 방식으로 알고리즘을 진행하게 된다.

트랜잭션 데이터 내에 T-알고리즘의 순서는 다음과 같다.

[Step 1] 전건부의 규칙 중 1의 개수가 1개인 것부터 N개인 것을 차례로 모아 정렬한다.

[Step 2] 트랜잭션 데이터베이스의 규칙 중 후건부의 값이 1인 전건부 규칙을 찾아낸다.

[Step 3] 정렬된 데이터끼리 비교하여 감축을 수행한다. 최소항과 비교해서 비교하는 항들 간에 서로 다른 값 최소항들을 골라낸다. 이때 결합한 값은 don't care bit (-)와 care bit(1,0)으로 표시된다. 예를 들어, 1의 개수가 한 개인 S_1은 0010이고 1의 개수가 두 개인 S_7이 1010일 때 {S_1, S_7}=-010으로 표시하여 1차 감축을 수행한다.

[Step 4] Step 3의 작업을 1의 개수가 n개인 항과 n+1인 항을 비교하여 감축한다.

[Step 5] 1차 감축 결과를 모아 Step 3과 Step 4의 동일한 방법으로 2차 감축을 시행한다.
단, 1차 감축에서 더 이상의 감축이 진행되지 않을 경우 2차, 3차 감축은 진행하지 않는다.
[Step 6] 2차 감축으로 중복되는 데이터 제거한다.

본 논문에서는 총 6 단계 감축 단계를 거쳐 연관 규칙을 생성하게 된다. 감축 순서는 Step 1에서 Step 6까지 이다. 생성된 n개의 연관규칙 집합들에 대하여 지지도와 신뢰도를 계산하여 최종 평가하게 된다.

4. 모의실험

본 논문에서 제안한 연관규칙의 알고리즘을 적용해 보기 위해 대학에서의 수강신청 과목에 관한 수강신청의 연관규칙을 추출해 보았다. 수강신청 데이터를 이용하여 학생들이 수강신청을 했을 때 각 과목간의 연관관계를 파악 할 수 있게 되고 추출된 정보를 이용하여 과목개설시 수강신청 인원을 추정할 수 있고 강의실 배정과 강사확보에 활용 될 수 있다.

표 2는 학생들의 교양과목 수강신청 데이터베이스 목록이다. 여기서 L은 전건부 과목, f는 후건부 과목을 나타내고 S는 수강 신청을 한 학생을 나타낸다.

즉, S_1이라는 학생이 L_3이라는 과목을 선택했을 때, f_1,f_2라는 과목을 동시에 신청했다는 규칙이 성립한다. 본 실험은 16명의 학생이 6개의 과목에 대한 수강신청 이력을 바탕으로 모의실험을 하였다.

표 2. 트랜잭션 데이터베이스
Table 2. Transaction Database

		X				Y	
	TID	L_1	L_2	L_3	L_4	f_1	f_2
	S_1	0	0	1	0	1	1
	S_2	0	0	0	1	1	1
	S_3	0	0	1	0	1	1
	S_4	0	1	0	0	1	0
1	S_5	1	0	0	0	1	0
1	S_6	0	1	0	1	0	1
	S_7	1	0	1	0	1	1
2	S_8	1	1	0	0	0	1
2	S_9	1	0	0	1	1	1
	S_{10}	0	0	1	1	1	0
3	S_{11}	0	1	1	0	1	0
3	S_{12}	1	1	0	1	0	1
	S_{13}	1	0	1	1	1	1
4	S_{14}	1	1	1	0	0	1
4	S_{15}	0	1	1	1	1	0
	S_{16}	1	1	1	1	1	0

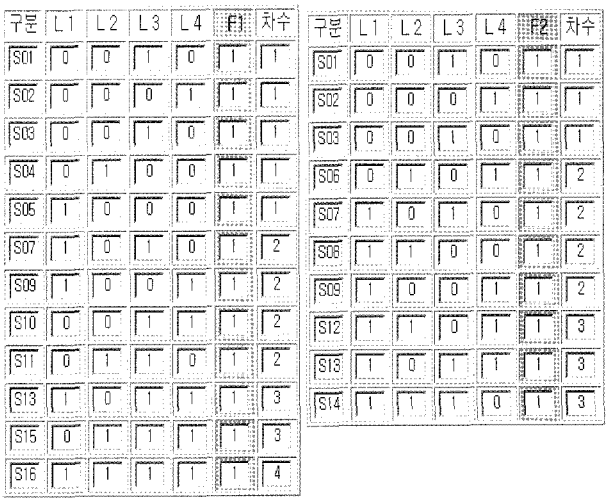
표 3는 표 2에 나타난 트랜잭션 데이터베이스에서 Apriori 알고리즘을 이용하여 추출한 연관규칙 중 $S_{\min}$(최소 지지도)이 20%이상인 10개의 연관규칙을 추출했다.

표 4에서 표 7까지는 T-알고리즘을 사용한 감축 과정을 나타내었다. 표 4은 표 2의 트랜잭션 데이터베이스에서 에서 Step 1,2의 방식으로 후건부 y의 값이 1인 데이터, 즉, $f_1, f_2 = 1$인 데이터를 추출해서 정렬한다.

표 3. $S_{\min}$ 20%이상인 연관규칙
Table 3. Association rules over $S_{\min}$20%

연관규칙 X→Y	Sup.
$L_1 \rightarrow f_1$	5(31%)
$L_2 \rightarrow f_1$	4(25%)
$L_3 \rightarrow f_1$	8(50%)
$L_4 \rightarrow f_1$	6(38%)
$L_1 \rightarrow f_2$	6(38%)
$L_2 \rightarrow f_2$	4(38%)
$L_3 \rightarrow f_2$	5(31%)
$L_4 \rightarrow f_2$	5(31%)
$L_3, L_4 \rightarrow f_1$	4(25%)
$L_3 \rightarrow f_1, f_2$	4(25%)

표 4. $f_1, f_2 = 1$인 데이터 추출
Table 4. If $f_1, f_2 = 1$, Reduction process



후건부 $f_1 = 1$일 때 Step 3,4의 방식으로 표 5와 같이 감축한다. 즉, {1,2},{2,3},{3,4}의 쌍으로 비교하여 감축한다.

표 5. $f_1=1$ 일 때 감축과정

Table 5. If $f_1=1$, Reduction process

1차 감축

구분	L1	L2	L3	L4	차수
S01S07	-	0	1	0	1
S03S07	-	0	1	0	1
S05S07	1	0	-	0	1
S02S09	-	0	0	1	1
S05S09	1	0	0	-	1
S01S10	0	0	1	-	1
S02S10	0	0	-	1	1
S03S10	0	0	1	-	1
S01S11	0	-	1	0	1
S03S11	0	-	1	0	1
S04S11	0	1	-	0	1
S07S13	1	0	1	-	2
S09S13	1	0	-	1	2
S10S13	-	0	1	1	2
S10S15	0	-	1	1	2
S11S15	0	1	1	-	2
S13S16	1	-	1	1	3
S15S16	-	1	1	1	3

2차 감축

구분	L1	L2	L3	L4	차수
S06S09S07S13	1	0	-	-	1
S01S10S07S13	-	0	1	-	1
S03S10S07S13	-	0	1	-	1
S05S07S09S13	1	0	-	-	1
S02S10S09S13	-	0	-	1	1
S01S07S10S13	-	0	1	-	1
S03S07S10S13	-	0	1	-	1
S02S09S10S13	-	0	-	1	1
S01S11S10S15	0	-	1	-	1
S03S11S10S15	0	-	1	-	1
S01S10S11S15	0	-	1	-	1
S03S10S11S15	0	-	1	-	1
S10S15S13S16	-	-	1	1	2
S10S13S15S16	-	-	1	1	2

3차 감축

구분	L1	L2	L3	L4	차수
S01S07S10S13	1	0	-	-	1
S01S10S11S15	-	0	1	-	1
S02S09S10S13	-	0	-	1	1
S03S07S10S13	-	0	1	-	1
S03S10S11S15	0	-	1	-	1
S05S07S09S13	0	-	1	-	1
S10S13S15S16	-	-	1	1	2

1차 감축 후 step 5의 방식으로 2차 감축을 수행하고 중복되는 데이터를 제거하므로 3차 감축까지 마무리한다.

표 6. $f_2=1$ 일 때 감축과정

Table 6. If $f_2=1$, Reduction process

1차 감축

구분	L1	L2	L3	L4	차수
S02S06	0	-	0	1	1
S01S07	-	0	1	0	1
S03S07	-	0	1	0	1
S02S09	-	0	0	1	1
S06S12	-	1	0	1	2
S08S12	1	1	0	-	2
S09S12	1	-	0	1	2
S07S13	1	0	1	-	2
S09S13	1	0	-	1	2
S07S14	1	-	1	0	2
S08S14	1	1	-	0	2

2차 감축

구분	L1	L2	L3	L4	차수
S02S09S06S12	-	-	0	1	1
S02S06S09S12	-	-	0	1	1

3차 감축

구분	L1	L2	L3	L4	차수
S02S06S09S12	-	-	0	1	1

위의 표 5와 동일한 감축 방법으로 표 6에서는 $f_2=1$ 일 때 세 번의 감축 과정을 거쳐 최종 1개의 규칙으로 감축되었다. 표 7은 $f_1 \cap f_2=1$ 일 때 감축과정이다.

표 7. $f_1 \cap f_2=1$ 일 때 감축과정

Table 7. If $f_1 \cap f_2=1$, Reduction process

$f_1 \cap f_2=1$

구분	L1	L2	L3	L4	F1	F2
S1	0	0	1	0	1	1
S2	0	0	0	1	1	1
S3	0	0	1	0	1	1
S4	1	0	1	0	1	1
S5	1	0	0	1	1	1
S6	1	0	1	1	1	1

1차 감축

구분	L1	L2	L3	L4	차수
S01S04	-	0	1	0	1
S03S04	-	0	1	0	1
S02S05	-	0	0	1	1
S04S06	1	0	1	-	2
S05S06	1	0	-	1	2

위의 경우 1차 감축 과정을 마치면 더 이상의 감축은 발생하지 않는다. 이유는 $\{S_1, S_7\}, \{S_9, S_{13}\}$을 비교했을 때, $-0--$라는 규칙이 된다. 즉, 비교 데이터가 3bit이상 차이가 나므로 규칙이 될 수 없다. 따라서 1차 감축으로 추출된 규칙이 최종 규칙이 된다.

표 8. 규칙감소율 비교

Table 8. Comparison of rule reduction rate

	Apriori 알고리즘	T-알고리즘		
규칙감소율	37%	$f_1=1$	$f_2=1$	$f_1 \cap f_2$
		56%	94%	69%

표 7과 같이 최종적으로 감축된 규칙의 감소율은 Apriori 알고리즘 37%, T-알고리즘 f의 상태에 따라 각각 56%, 94%, 69%로 나타났다.

연관규칙의 성립을 위해 한 가지 예를 들면, $X=L_3$이고 $Y=f_1, f_2$,이라 가정했을 때 Apriori 알고리즘을 이용하여 연관규칙 X→Y의 신뢰도와 지지도는 다음과 같이 구할 수 있다.

$$S=\frac{X \cup Y}{N}=\frac{4}{16}=25\%,\quad C=\frac{X \cup Y}{X}=\frac{4}{9}=44\%$$

이고, T-알고리즘을 사용하여 감축된 규칙에서의 연관규칙 X→Y의 지지도와 신뢰도는

$S=\frac{X \cup Y}{N}=\frac{3}{5}=60\%$, $C=\frac{X \cup Y}{X}=\frac{3}{3}=100\%$이다.

따라서 TID S_1, S_7, S_{13}인 학생이 L_3라는 과목을 선택했을 때, f_1, f_2라는 과목을 선택할 지지도와 신뢰도는 각각 60%와 100%로 나타났다.

따라서 Apriori 알고리즘을 사용했을 때와 제안 알고리즘을 사용해서 규칙을 감소했을 때를 비교하면 제안 알고리즘을 사용했을 경우 규칙을 보다 효율적으로 감축할 수 있었고 지지도와 신뢰도도 함께 증가시킬 수 있다는 것을 알 수 있다.

5. 결론 및 향후연구

연관규칙 마이닝에서는 방대한 수의 트랜잭션데이터를 다루기 때문에 연관규칙 탐사기법에서 규칙을 도출하는 데는 많은 시간이 소요된다.

본 논문에서는 기존의 연관성 탐사기법과는 다른 방법인 T-알고리즘을 적용하고 대학에서의 수강신청 시스템에 적용시켜보았다.

그 결과 제안된 T-알고리즘을 이용하여 트랜잭션데이터베이스에 존재하는 규칙들을 효과적으로 감축시킬 수 있고 항목간의 지지도와 신뢰도도 향상 시킬 수 있었다. T-알고리즘을 적용한 수강신청 시스템을 이용함으로 강의관련 학사 업무에 효과적으로 대처 할 수 있을것이다.

향후 연구로는 다중치의 개념을 적용한 웹 마이닝에서의 연관규칙 감축 방법에 대한 연구가 요구된다.

참 고 문 헌

[1] I. Witten, E.Frank, *data Mining*, Morgan Kaufmann Publisher, 2000.

[2] Argrawal, R.,Imielinski,T. and Swami, A. " Mining Association Rules in Large Databases," *In Proc. Int'l Conf. on Management of Data,* ACMSIGMOD, Washington D.C, pp.207-216, May. 1993 .

[3] 정환묵 , *소프트컴퓨팅*, 내하출판사, 2008.

[4] 강용성, 김미선, 서재현,"Fast-Apriori 알고리즘을 이용한 이상행위 탐지 프로파일링 연구", *한국인터넷정보학회 학술발표대회 논문집*,pp.483~486, 2003.

[5] Pang-ning Tan, M.Steinbach,Vipin Kumer, *Introduction to data mining*, Publisher Addison-Wesley, 2006.

저 자 소 개

박진희(Jin-Hee Park)

2002년 : 대구가톨릭대학교 컴퓨터정보통신공학부 학사

2003년 : 대구가톨릭대학교 전산통계학과 석사

2006년 : 대구가톨릭대학교 컴퓨터정보통신공학 박사과정수료

2008년~現 : 대구가톨릭대학교 강의전담교수

관심분야 : 퍼지이론, 데이터마이닝

E-mail : aimajor@cu.ac.kr

정환묵(Hwan-Mook Chung)

10권 4호 참조

Phone : +82-53-850-2741

Fax : +82-53-850-2741

E-mail : hmchung@cu.ac.kr