시간 연관규칙의 유지를 위한 점진적인 마이닝 기법

백옥현⁰ 이준욱 김영균 류근호
충 북 대 학 교 데 이 터 베 이 스 연 구 실
{ohpaek, junux, ykkim, khryu}@dblab.chungbuk.ac.kr

An Incremental Mining Technique for Maintenance of Temporal Association Rules

Ok Hyun Paek⁰ Jun Wook Lee Young Kyun Kim Keun Ho Ryu
Database Laboratory, Chungbuk National University

요 약

실세계의 여러 응용에서 데이터베이스의 크기는 계속적으로 증가되어 왔으며, 이러한 데이터베이스 내에서 유용한 지식을 찾아내기 위한 다양한 연구가 진행되어 왔다. 데이터베이스는 시간이 흐름에 따라 동적으로 변화된다. 현재의 연구는 이러한 데이터베이스에서 효과적으로 규칙을 발견하는데 초점이 모아지고 있다. 그러나, 이런 변화에 따라서 기존에 발견되었던 규칙들은 더 이상 유효하지 않을 수 있기 때문에 이전에 발견되었던 규칙들은 유효한지 검증되어야 한다. 데이터베이스가 증가할 때마다 전체를 다시 탐색해서 규칙을 찾는 것은 효과적인 방법이 아니므로, 점진적으로 규칙을 유지할 수 있는 알고리즘이 필요하다. 이 논문에서는 이전에 발견되었던 규칙이 물리적으로 저장되고 그 후에 데이터베이스가 업데이트된 것을 고려하여 규칙, 특히 시간 연관규칙을 점진적으로 유지할 수 있는 기법을 제시한다.

1. 서론

데이터 마이닝은 지식탐사라고도 하며 누적되어있는 원시 데이터를 변환, 분류하고 그 규칙성 및 패턴 등을 자동적으로 수행하여 획득해 냄으로써 전략적인 의사결정을 지원할 수 있는 기술이다.

데이터 마이닝의 여러 기법 중 연관규칙은 데이터 마이닝에서 가장 활발하게 연구가 된 분야중의 하나이다. 연관규칙은 '어떤 사건이 일어나면 다른 사건이 일어난다'와 같은 사건 간의 연관성을 말한다[1]. 주어진 트랜잭션의 집합이 있고 서로소 관계인 두개의 부분집합의 연관규칙은 'X→Y'로 표시할 수 있다. 이 때, 조건부 항목집합 X는 결론부 항목집합 Y를 야기한다고 해석할 수 있다. 그러나, 이러한 기존의 연관규칙은 트랜잭션 데이터가 가지고 있는 시간적인 특성을 간과함으로써 의미있는 시간 관련 규칙들을 찾을 수 없었다. 예를 들어, 데이터베이스가 1990년 1월 1일과 2000년 12월 31일 사이의 트랜잭션을 포함하고 있다면, 연관규칙 X→Y는 이 데이터베이스 전체에 대해서는 최소지지도와 최소신뢰도를 만족하지 않을 수 있다. 그러나, 각각 다른 여러 시간 간격을 고려한다면 다음과 같은 시간 연관규칙을 찾을 수 있다.

- 1993년 1월과 1995년 12월 동안에 규칙 X→Y는 최소지지도와 최소 신뢰도를 만족한다.
- 70%의 가능성을 갖고 규칙 X→Y는 매해 여름동안 최소지지도와 최소신뢰도를 만족한다.
- 90%의 가능성을 갖고 규칙 X→Y는 1997년부터 지금까지의 가을동안 최소지지도와 최소신뢰도를 만족한다.

위의 예는 데이터 마이닝을 통해서 지식을 얻고자 하는 사용자가 전체 데이터 집합에 대한 규칙을 얻기보다는 특정 기간동안만의 시간적인 규칙을 찾아냄으로써 더 유용한 지식을 얻을 수 있다는 것을 의미한다.

일반적으로, 실세계의 도메인에서 데이터베이스는 시간에 따라 동적으로 변화한다. 그러나 데이터베이스가 갱신될때마다 전체 데이터베이스에 대해 다시 규칙을 탐사하는 것은 데이터가 대량인 환경에서는 적합하지 않은 방법이다. 따라서, 발견된 규칙을 점진적으로 유지하고 갱신할 수 있는 방법이 필요하다. 점진적인 데이터마이닝 기법을 사용하면 데이터베이스의 스캔을 줄여서 규칙을 찾는데 드는 비용을 최소화할 수 있고, 알고리즘의 효율성을 극대화하면서 발견된 규칙의 질을 유지시킬 수 있다.

연관규칙에 대한 점진적인 마이닝 기법에 대한 연구는 진행되었지만[4,5], 시간 연관규칙에 대한 점진적인 유지 기법에 대해서는 아직 연구가 이루어지지 않았다. 이 논문에서는, 아직 일반적인 정의가 내려져 있지 않은 시간 연관규칙에 대해 기술한다. 또한 이러한 시간적인 특성을 지니는 연관규칙에 대해 이전 단계에서 발견되었던 주요항목집합을 이용하여 점진적으로 유지할 수 있는 방법을 제시한다.

2. 관련연구

2.1 시간 연관규칙

연관규칙은 트랜잭션의 집합에서 항목들간의 연관관계를 발견하는 기법이다. 항목집합 I가 있을 때, I의 부분집합인 X, Y에 대한 연관규칙 X→Y는 X가 나타낼 때 Y가 나타난다는 두 항목집합의 연관성을 나타낸다. 이 때, 이 규칙은 최소 지지도와 최소 신뢰도를 만족해야 한다.

시간 연관규칙은 위와 같은 정적인 연관규칙에 시간적인 개념을 도입한 것으로서, 아직까지 명확하게 일관된 정의는

내려지지 않았다. Chen은 [3]에서 시간 데이터베이스에서 시간 연관규칙을 탐사하기 위한 알고리즘과 마이닝 언어, 시스템의 구조를 제시하였다. 여기에서의 시간 연관규칙은 기존의 연관규칙에 시간적인 표현을 고려해서 달력 시스템에 의존한 특정 시점이나 시간 간격의 데이터를 분석하였다. Ale 등은 [2]에서 주요항목집합을 발견할 때, 연관규칙에 존재하는 항목들의 생명시간을 고려한다. 즉, 항목이 존재하는 기간동안만의 지지도인 시간 지지도라는 임계치를 두어서 시간에 따른 상대적인 지지도를 고려할 수 있게 하였다.

2.2 점진적인 데이터 마이닝

Cheung 등은 [4]에서 데이터베이스에 새로운 트랜잭션 데이터가 삽입 되었을 때, 기존에 발견되었던 연관규칙을 효과적으로 유지하고 갱신하기 위한 FUP 알고리즘을 제안하였다. FUP 알고리즘에서는, 이전에 발견되었던 주요항목이 지지도와 함께 데이터베이스에 저장되어 있다고 가정하고, 새로운 트랜잭션이 삽입되면 먼저 삽입된 데이터베이스에 대해서만 주요항목을 구한다. 삽입되기 이전의 데이터베이스는 필요한 경우에만 검색을 수행하게 됨으로써 규칙을 찾는 비용을 줄일 수 있는 알고리즘을 제시한다. [5]에서는 갱신되기 이전의 데이터베이스에 대한 스캔 비용을 줄이기 위해 두 가지의 지지도 임계치를 사용하여 Pre-large 항목의 개념을 제안하고, 그것에 기반한 점진적인 데이터마이닝 알고리즘을 제안하였다.

점진적인 기법 자체가 시간적인 특성을 갖기 때문에, 기존의 정적인 데이터 마이닝 기법에 적용하는 것보다 시간적인 특성을 가진 규칙에 적용했을 때 더욱 효과적일 것으로 기대된다.

3. 시간 연관규칙

T는 시간 도메인이라고 하는 시간 간격의 집합이다. $I=\{i_1, i_2,\dots,i_l\}$ 은 항목의 집합이다. D는 시간값을 가진 트랜잭션의 집합이다. 각각의 트랜잭션 S의 투플은 <tid, itemset, timestamp>을 갖는다. 이 때, S.tid는 트랜잭션 식별자, S.itemset은 S.itemset $\subset$ I인 항목들의 집합이며, S.timestamp는 S.timestamp $\subset$ T인 트랜잭션 S의 시간을 나타낸다. 항목의 집합 $X \subset I$는 항목집합이라 한다. 항목집합에 있는 항목들의 개수는 그 항목집합의 크기라 정의한다. 크기가 k인 항목집합은 k-항목집합이라 한다. $X \subseteq$ S.itemset이면 트랜잭션 S는 항목집합 X를 포함한다고 한다.

P를 시간 표현의 집합이라고 하자. P의 나열인 $\Phi(P) = \{p_1, p_2,\dots p_n\}$은 시간 표현 P에 대한 시간간격의 집합을 나타낸다. 마이닝을 수행하는 응용들에 있어서 시간 도메인이 유한하므로, $\Phi(P)$도 유한한 집합이다. 여기서는 p_i에 속하는 시간값을 가진 트랜잭션이 있는 D의 부분집합을 $D(p_i)$라 표기한다. 이 논문에서는 시간표현은 [3]에서 제시한 달력 시간 표현을 사용하였다.

정의 1. 시간 연관규칙은 <R, P>의 쌍으로 이루어진다. R은 $X \subset I$, $Y \subset I$ 이고 $X \cap Y = \emptyset$인 $X \rightarrow Y$의 형식을 가진다. P는 시간 표현을 나타낸다. 이것은 P로 표현되는 시간의 간격동안의 트랜잭션에서 한 트랜잭션 내에 X 항목집합이 있으면 Y 항목집합이 있다는 규칙을 나타낸다.

다음은 시간연관 규칙의 예를 보인다.

예 1. <milk,bread → cheese, T_DOM during [Years(1997) • Months(3), Years(1998) • Months(7)]>

이것은 1997년 3월부터 1998년 7월 동안 우유와 빵을 산 사람은 치즈를 구입한다는 연관성을 나타낸다.

예 2. <hiking-boots → outerwear, Years • Months (4:6) during [Years(1994), Years(1998)]>

이것은 1994년부터 1998년 매해 봄동안 하이킹 부츠를 구입하는 고객은 외투를 구입한다는 연관성을 표현한다.

예 3. <Diaper → Beer, Days • Hours (19:20) during [Years(1998) • Months(7) , +∞]>

이것은 1998년 7월 이래로 현재까지 매일 저녁시간에 기저귀를 구입하는 고객은 맥주를 구입한다는 패턴을 표현한다.

정의 2. 시간 도메인 T에 걸친 시간값을 가진 트랜잭션의 집합 D와 시간 표현 집합 P가 주어졌을 때, 다음과 같이 정의한다.

(1) 규칙 <X → Y, P>는 $D(p_i)$에서 $X \cup Y$를 포함하는 트랜잭션이 s% 이상이면 시간 간격 p_i, $p_i \in P$ 동안 지지도 s%를 가진다.

(2) 규칙 <X → Y, P>는 $D(p_i)$에서 X를 포함하는 트랜잭션의 c%이상이 Y를 포함하면 시간 간격 p_i, $p_i \in P$ 동안 신뢰도 c%를 가진다.

(3) 규칙 <X → Y, P>는 $\Phi(P)$내에 있는 시간 간격의 f% 이상 동안 규칙 <X → Y, P>가 지지도 s%와 신뢰도 c%를 가지면, 데이터베이스 D의 빈도 f%에 대해 지지도 s%와 신뢰도 c%를 가진다.

정의 3. 시간도메인 T에 걸친 시간값을 가진 트랜잭션의 집합 D와 시간 표현 집합 P가 주어졌을 때, 시간 연관규칙을 탐사하는 것은 D에서 사용자가 지정하는 최소지지도 minsup과 최소신뢰도 minconf, 최소빈도 minfreq를 만족하는 <X → Y, P> 형태의 규칙을 찾는 것이다.

이 논문에서는 [1]에서 제시한 주요항목집합의 개념을 다음과 같이 확장해서 사용한다. $|D(p)|$는 p시간간격 동안 데이터베이스 D의 트랜잭션의 수를 나타낸다.

정의 4. 트랜잭션 데이터베이스 D에서 시간 간격 p동안 항목집합 X의 지지도 $X.supp_{D(p)}$는 D(p)상에서 항목집합을 포함하는 트랜잭션의 수이다. 항목집합 X는 $X.supp_{D(p)} / |D(p)| \geq minsup$ 이면 시간 간격 p동안 주요하다고 한다. 항목집합 X의 빈도 $X.freq_{D(P)}$는 $\Phi(P)$의 시간 간격에서 X가 나타나는 시간간격의 수이다. 항목집합 X는 $X.freq_{D(P)} / |\Phi(P)| \geq minfreq$이면 시간 표현 P에 대해 주요하다고 정의한다.

4. 시간 연관규칙을 위한 점진적인 마이닝 기법

4.1 시간 연관규칙의 갱신

앞서 들었던 시간 연관규칙의 예 1,2는 시간 도메인 상에서의 절대적인 시간 간격 내에서의 규칙을 탐사하기 때문에 시간이 지나도 규칙은 변함이 없다. 그러나 예 3의 경우, 시간의 흐름에 따라 시간 도메인이 증가하므로 규칙이 유효한지 검증해야 할 필요가 있다.

증가된 시간 도메인을 ΔT라 하고, ΔT가 증가한 만큼 P도 증가하게 된다. 증가된 P를 ΔP라 하면 ΔP의 나열인 $\Phi(\Delta P)=\{\Delta P_1, \Delta P_2,\dots, \Delta P_m\}$이 된다. 또한 증가된 데이터베이스는 ΔD로 표기한다.

정의 5. 항목집합 X는 $X.freq_{D(P)} + X.freq_{\Delta D(\Delta P)} / |\Phi(P)| + |\Phi(\Delta P)| \geq minfreq$ 이면 시간표현 $P \cup \Delta P$에 대해 주요하다고 정의한다.

데이터베이스 D에서 P에 대해 주요한 항목을 L_k^D라 하면, 시간 연관규칙을 갱신하는 문제의 핵심은 $D \cup \Delta D$에서

시간표현 $P \cup \Delta P$에 대해 주요한 항목집합 $L_k^{D \cup \Delta D}$을 찾는 것이다. 또한 시간 도메인에 대해 더해진 데이터베이스는 그 자체로서 유용한 규칙을 포함할 수 있다. ΔD의 데이터는 최근의 새로운 변화를 반영한다. 이것은 $L_k^{\Delta D}$를 찾음으로써 얻어질 수 있다.

이 논문에서는 이전에 있었던 데이터베이스에 삽입만 수행되었고, 알고리즘의 인수인 minsup과 minfreq가 동일하며, 이전 과정의 주요 항목집합이 빈도값과 시간 정보를 가지고 데이터베이스에 저장되어있다는 가정 하에서 알고리즘을 기술한다.

4.2 점진적인 시간 연관규칙 유지 알고리즘

이 절에서 설명되는 점진적인 시간 연관규칙 유지 알고리즘은 Agrawal에 의해 제시되었던 Apriori 알고리즘[1]을 기반으로 한다. 제시하는 알고리즘은 기존에 있었던 규칙의 질을 검증하는 부분과, 새로이 나타나고 있는 경향을 분석하는 부분에 초점을 두었다.

제시하는 알고리즘은 크게 세 단계로 나누어진다.

단계 1. 증가된 데이터베이스 ΔD를 스캔해서 각각 다른 시간 간격에 존재하는 후보 1-항목집합의 발생수와, 빈도수를 센다.

단계 2. ΔD의 후보 1-항목집합에서 minfreq를 만족하는 주요 1-항목집합을 생성하고, ΔD의 후보 1-항목집합과 D의 주요 1-항목집합을 비교해서 전체 minfreq를 만족하는 전체 주요 1-항목집합을 생성한다.

단계 3. 각 ΔD의 주요 (k-1)-항목집합과 $D \cup \Delta D$의 주요 (k-1)-항목집합은 ΔD와 $D \cup \Delta D$ 각각의 후보 k-항목집합을 만들기 위해 사용된다. 후보집합 생성은 [1]에서 제시한 apriori-gen() 함수가 사용된다. ΔD가 스캔되면서 각 k-항목집합의 지지도와 빈도가 세어지고, 각각의 후보집합에서 minfreq를 만족하는 항목집합이 선택된다. 이 과정은 새로운 주요항목집합이 찾아지지 않을때까지 반복된다.

다음은 점진적인 시간 연관규칙을 위한 단계별 알고리즘을 보여준다.

입력 : (1)D:원래 데이터베이스, (2)ΔD:증가분 데이터베이스, (3)$\Phi(\Delta P)$:증가된 시간간격의 집합, (4)L_k^D:D에서 P에 대한 주요 항목집합, (5)minsup:최소지지도, (6)minfreq:최소빈도

출력 : (1)$L_k^{D \cup \Delta D}$: $D \cup \Delta D$의 $P \cup \Delta P$에 대한 모든 주요항목, (2) $L_k^{\Delta D}$: ΔD의 ΔP에 대한 주요항목

단계

```
C1^ΔD={1-후보항목집합} /*ΔD의 각각 다른 시간간격에
    존재하는 1-후보항목집합*/
foreach transaction S∈ΔD(Δp) do
    count support, frequent for each Δp
L1^ΔD = {c∈C1^ΔD | c.freq_ΔD(ΔP)/|Φ(ΔP)| ≥ minfreq}
foreach X∈ L1^D do
    if X.freq_D(p)+ X.freq_ΔD(Δp)/|Φ(P)|+|Φ(ΔP)| ≥
     minfreq then
        L1^D∪ΔD = L1^D∪ΔD ∪ {X};
for (k=2;L_k-1^D∪ΔD ≠∅ and L_k-1^ΔD ≠∅;k++) do begin
    Ck^ΔD= apriori-gen(L_k-1^ΔD);
    Ck^D∪ΔD= apriori-gen(L_k-1^D∪ΔD);
    foreach transaction S∈ΔD(Δp) do begin
        foreach candidate c∈Ck^ΔD(p) do
            count support, frequent for each Δp
        foreach candidate c∈ Ck^D∪ΔD do
            count support, frequent for each Δp
    end
    Lk^ΔD={c∈Ck^ΔD| c.freq_ΔD(ΔP) / |Φ(ΔP)| ≥ minfreq}
    Lk^D∪ΔD = {c∈Ck^D∪ΔD | c.freq_D(P) + c.freq_ΔD(ΔP) /
        |Φ(P)| + |Φ(ΔP)| ≥ minfreq}
end
```

예 4. 데이터베이스 D가 증가되었고, $|\Phi(P)|=40$, $|\Phi(\Delta P)|=10$이라고 가정하자. minfreq=75%이고 minsup=20%이고, I={A, B, C, D, E}이다. L_1^D = {A, B, C, D}이고, 각각의 빈도는 $A.freq_{D(P)}=35$, $B.freq_{D(P)}=33$, $C.freq_{D(P)}=32$, $D.freq_{D(P)}=37$이다. ΔD에 대해 스캔한 결과, $C_1^{\Delta D}$ = {A, B, C, E}이고, 각각의 빈도는 $A.freq_{\Delta D(\Delta P)}=5$, $B.freq_{\Delta D(\Delta P)}=8$, $C.freq_{\Delta D(\Delta P)}=5$, $E.freq_{\Delta D(\Delta P)}=9$ 라고 가정하자. ΔD에서의 minfreq인 10*75%=8 이상의 빈도를 가지는 B와 E가 선택된다. 따라서 $L_1^{\Delta D}$={B, E}이다. 한편, $X \in L_1^D$인 항목집합의 빈도와 $C_1^{\Delta D}$ 를 빈도를 비교하면, $X.freq_{D(P)}$ + $X.freq_{\Delta D(\Delta P)}$ 가 (40+10)*75% 은 A, B 항목이 선택되어 $L_1^{D \cup \Delta D}$={A, B}가 되어 C, D는 제거된다.

$L_1^{\Delta D}$에 대한 후보집합 $C_2^{\Delta D}$={BE}가 생성된다. 또한 $L_1^{D \cup \Delta D}$를 이용하여 $C_2^{D \cup \Delta D}$={AB}가 생성된다. ΔD가 스캔되어 $BE.freq_{\Delta D(\Delta P)}=8$, $AB.freq_{\Delta D(\Delta P)}=5$라고 가정하자. BE는 ΔD에서의 minsup을 만족하여 $L_2^{\Delta D}$={BE}가 된다. L_2^D = {AB, AC, BD}라고 하자. AC, BD는 부분집합이 $L_1^{D \cup \Delta D}$에 포함되어 있지 않으므로 고려사항이 아니다. $AB.freq_{D(P)}=33$이라 하면, $AB.freq_{D(P)}$ + $AB.freq_{\Delta D(\Delta P)}$는 38이 되어 minfreq인 ($|\Phi(P)|+|\Phi(\Delta P)|$) * 75% 이상이므로, $L_2^{D \cup \Delta D}$={AB}이다.

5. 결론 및 향후 연구

이 논문에서는 연관규칙에 시간적인 요소를 고려한 시간 연관규칙에 대해 정의하고, 시간연관규칙의 효율적인 유지를 위한 점진적인 알고리즘을 제시하였다. 제시된 알고리즘은 이전 단계에서 발견되었던 규칙의 정보를 이용하여, 시간 연관규칙에 점진적인 데이터 마이닝 기법을 적용함으로써 규칙의 유효성을 검증할 수 있도록 하였다. 또한 시간 도메인에 대해 추가된 데이터베이스에 대해서만 규칙 탐사를 수행함으로써 새로운 비즈니스 전략이나 고객의 구매패턴 변화 등을 낮은 비용으로 빠르게 분석할 수 있다. 향후, 이 논문에서 제시한 알고리즘의 구현과, 능동적인 시간 연관규칙의 마이닝 기법에 대한 연구가 필요하다.

6. 참고문헌

[1] R.Agrawal, R.Srikant, Fast Algorithm for Mining Association Rules, Proc. of VLDB'94, pp.478-499.

[2] J.M.Ale and G.H.Rossi, An Approach to Discovering Temporal Association Rules, Proc. of SAC'00, pp.294-300.

[3] X.Chen, I.Petrounias and H.Heathfield, Discovering Temporal Association Ruels in Temporal Databases, Proc. of IADT'99.

[4] D.W.Cheung, J.Han, V.Ng, and C.Y.Wong, Maintenance of discovered association rules in large databases : An incremental updating techque, Proc. of ICDE'96.

[5] T.-P.Hong, C.-Y.Wang and Y. -H. Tao, Icremental Data Mining Based on Two Support Thresholds, Proc. of Knowledge-Based Intelligent Engineering and Allied Technologies, 2000.

[6] 서성보, 점진적 시간 데이터마이닝 기법 구현, 충북대학교 대학원 전자계산학과 이학석사 학위논문, 2001.