시계열 데이터베이스와 강결합된 규칙발견 알고리즘 설계와 구현

박인창[0] 김성규
안양대학교 컴퓨터학과
icpark@amadeus.yonsei.ac.kr, sgkim@aycc.anyang.ac.kr

Design and Implementation of Rule Discovery Algorithm strongly coupled with Time-series databases

Inchang Park[0], Seonggyu Kim
Dept of Computer Science, Anyang Unviersity

요약

마이닝 시스템은 그 특성에 따라 매우 다른 형태의 구현 방법이 존재한다. 그러므로 마이닝 시스템간 호환성이나 재사용성은 매우 낮다. 본 논문에서는 이 문제를 시계열 데이터베이스를 통한 RDB와 강 결합함으로써 표준화에 대한 문제를 해결하고자 시도하였다. RDB와의 강 결합은 표준화 문제를 해결함과 더불어 마이닝 시스템에 DBMS의 관련 기술을 이용함으로써 성능을 극대화시킨다. 특히 DBMS의 인덱스 기능을 이용함으로써 마이닝 시스템의 성능 향상을 시도하였다. 본 논문에서는 기존의 순차패턴 탐사의 시간개념 부재, 트랜잭션 데이터베이스 기반구조, 그리고 알고리즘 수행에 있어서 메모리 한계에 따른 문제등의 단점을 지적하고, 이를 수정하고 보완하기 위해서 시간 거리와 패턴 길이의 개념을 확장하였으며 그에 따른 연관규칙의 관련 공식을 수정 보완하여 제안한다. 또한 RDB와의 강 결합되어 기존의 트랜잭션 데이터베이스 구조를 벗어나 시계열 데이터에 보다 쉽게 적용할 수 있는 절차와 알고리즘을 제안한다.

1. 서론

미리 구축된 데이터베이스의 효과적인 활용을 위한 분야인 데이타 마이닝에 대한 연구가 최근 몇 년 동안 많이 이루어지고 있다. 거대한 데이터베이스에서 유용한 정보를 발견하는 데이터 마이닝은 인공지능에 원천을 둔 알고리즘과 데이터베이스 기술이 결합되어 발전하고 있다. 이러한 알고리즘들은 데이터 형태나 원하는 결과에 따라 적절하게 선택하여 활용된다. 시계열 데이터는 일반 기업에서 많이 활용되고 있어 데이터 마이닝에서 중요하게 다뤄야할 데이터이다. 일반적으로 시계열데이터에 대한 마이닝은 데이터에 대한 윈도우[1]내에서의 비슷한 패턴 (pattern or sequence)를 찾는 방법을 사용한다. 따라서 시계열 데이터는 일반적인 데이터 마이닝 알고리즘을 적용하기 곤란하므로 관련논문들은 시계열 데이터에 적절히 적용될 수 있는 마이닝 알고리즘을 발표했다. 본 논문에서도 시계열 데이터를 위한 알고리즘을 제안 한다. 하지만 본 논문은 연관규칙에서 순서개념의 확장인 순차패턴탐사를 기준으로 출발하여 시계열 데이터를 위한 알고리즘을 설계, 구현 한다.

본 논문의 구성은 2장에서는 순차패턴 탐사를 바탕으로 수정되어진 알고리즘을 제안한다. 3장에서는 제안된 알고리즘을 실제 적용하고 그 성능을 평가하며, 4장에서는 결론과 함께 앞으로의 연구 방안에 대해서 기술한다.

1) 테스트에 사용되어진 각각의 데이터에 대한 시간의 거리

2. 시계열 데이터를 위한 알고리즘 설계

2.1 순차패턴 탐사의 문제점[2]

순차패턴 탐사는 다음과 같은 문제점을 내포하고 있다.

(1) 시간의 개념

AprioriAll은 항목들간에의 순서는 알 수 있으나 시간의 거리를 알지는 못한다.

(2) 메모리 한계

대부분의 데이터 마이닝 논문들은 알고리즘 수행에 따른 메모리 관리에 대해 다루고 있지 않다. 실험은 편의상 데이터를 모두 메모리에 올려 수행한다. 따라서 많은 메모리가 필요하고 결과적으로 많은 데이터를 처리할 수 없다.

(3) 트랜잭션 기반 데이터베이스

기본으로 주어진 고객 데이터베이스와 비슷한 구조를 갖는 데이터형태에 적합하다. 데이터를 트랜잭션 데이터베이스 형태로 변환하는 것도 가능하나, 그렇지 못한 것도 있다. 예를 들면 과학적인 데이터를 들 수 있다.

2.2 시계열 데이터를 위한 알고리즘 설계

2.2.1 가정 및 정의

본 논문에서는 시계열 데이터베이스 D를 다음과 같이 가정한다. 데이터베이스 D의 구성은 기본적으로 타임스탬프(time-stamp) 속성을 포함하고 있어야 한다. 데이터베이스 D는 타임스탬프를 키로 오름차순 정렬되어 있으며 타임스탬프의 값은 데이터베이스 D에서 행을 유일하게 구분할 수 있어야 한다.

[정의 1]

항목집합(Itemset) I은 데이터베이스 D에 들어있는 항목들의 집합이다. 예를 들어 I = {A, B, C, D} 이면 데이터베이스 D에는 항목 E가 나타나지 않는다. |I|는 항목의 개수이다. 이 예에서 |I| = 4이다.

[정의 2]

X는 패턴의 가정이다. X는 I의 항목들로 구성된다. Y는 규칙의 결과이며 Y도 I의 항목으로 구성된다. 본 논문에서, Y는 단 하나의 항목을 가지는 결과로 규정한다. 본 논문에서는 기존의 연관규칙과 달리 X∩Y=∅이 아니다

[정의 3]

l은 X의 길이(length)이며 window의 크기로 봐도 관계 없다. minL ≥ 0이며, minL ≤ l ≤ maxL을 만족한다.

[정의 4]

d는 X와 Y사이의 시간적 거리(distance)이다. minD ≥ 0이며, minD ≤ d ≤ maxD를 만족한다.

[정의 5]

|X|는 X의 경우의 수이다. $|X| = |I|^l$을 만족한다. |Y|는 Y의 경우의 수이다. Y의 경우의 수는 X와 같이 $|Y| = |I|^l$이다.

[정의 6]

startP(X)는 데이터베이스 D에서의 X의 시작위치(position)이며 endP(X)는 X의 마지막 위치이다. endP(X) = startP(X) + l - 1 이다.

[정의 7]

Group $G_{l,d}$은 각각의 같은 거리와 길이를 가진 X의 집합을 말한다. Group $G_{l,d}$는 지지도와 신뢰도 계산을 위해 사용된다. 즉 같은 거리와 길이를 가진 Group간에서 지지도와 신뢰도를 계산한다. $|G_{l,d}|$는 $G_{l,d}$ 에 적용되는 데이터 총 개수이다.

[정의 8]

|R|은 만들어질 규칙의 예상 개수이다. |P|는 실제로 만들어진 규칙의 개수이다. |R| ≥ |P|을 만족한다. |P|는 데이터 집합과 관련있다. 예를 들어 데이터 집합의 크기 0이라면 |R|과 관계없이 |P| = 0이 된다. |R|은 다음과 같이 계산된다.

$$|R| = \sum_{i=\min L}^{\max L} |I|^{i+1}(\max D - \min D + 1)$$

[정의 9]

|A|는 데이타베이스에 실제적으로 접근된 규칙의 수이다. |U|는 수정된 규칙의 수이다. 즉,

$$|A| = |P| + |U|$$

|A| = |P|일 경우는 모든 규칙이 중복되지 않고 한번만 나온 경우가 되고 이때 |U| = 0이 된다.

[정의 10]

지지도는 $G_{l,d}$ 중 X가 속할 확률이다. 연관규칙에서의 지지도는 다음과 같다.

$$\text{지지도} = \frac{G_{l,d}\text{에 X가 나타나는 데이터 개수}}{|G_{l,d}|} \times |l|^{-1} \times \frac{|P|}{|R|}$$

[정의 11]

신뢰도는 Gl,d에 속하는 X의 endP(X) + d에 Y가 나올 확률이다.

$$\text{신뢰도} = \frac{\text{endP(X) + d에 Y가 나타날 확률}}{G_{l,d}\text{에서 X가 나타날 개수}}$$

[정의 12]

R은 규칙이다. R은 다음과 같이 표현된다.

$$R : X_l \xrightarrow{d} Y \text{ (지지도, 신뢰도)}$$

2.2.2 규칙탐사 알고리즘 설계

순차패턴탐사를 포함한 기존의 메모리 의존적 알고리즘의 문제점을 수정보완하기 위한 알고리즘을 제안한다.

[표 1] 규칙탐사 알고리즘

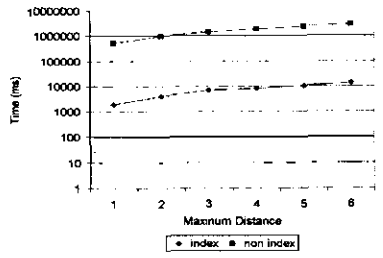
```
data[# of record] := data stored in database
k := 0
while ( k < # of record )
begin
    l := Minimum of length
    while ( l ≤ Maximum of length )
    begin
        startPX = k
        endPX = k + l - 1
        X := extract from data[startPX : endPX]
        d := Minimum of distance
        while ( d ≤ Maximum of distance )
        begin
            PositionY = endPX + d
            Y := extract data[PositionY]
            if R : X → Y is already existed
            then
                ruleUpdate ( X, Y, d, l )
            else
                ruleInsert ( X, Y, d, l )
            end if
            d := d + 1
        end
        l := l + 1
    end
    k := k + 1
end
```

3. 알고리즘 실험 결과

3. 1 인덱스의 필요성

규칙이 데이터베이스와 강 결합된 탐사를 하면서 삽입과 수정을 계속 하게된다. 삽입과 수정이 빈번한 곳에서 인덱스의 구성은 바람직하지 않지만, 검색도 많은 시간을 소

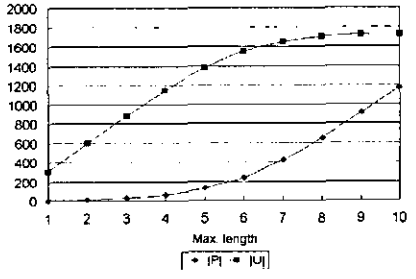
요함으로 전체적으로 효율성은 높아진다.



[그림 3] 인덱스 적용 결과 (B^+ 트리)

실험은 maxD을 1에서 6까지 늘려가며 결과를 도출했다. 실험결과 거리와 길이 모두, 인덱스를 적용한 경우가 약 10^2배정도 빠른 것으로 나타났다.

3. 2 maxL에 따른 |U|와 |P|

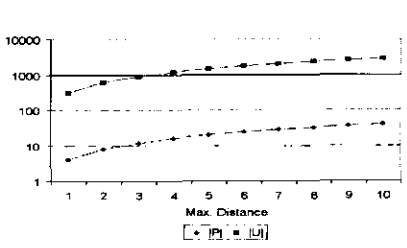


[그림 4] maxL에 의한 |P|와 |U|

그림 4에서 |P|는 |U|보다 작지만 maxL이 더욱 커질수록 |P|>|U|가 될 것임을 짐작 할 수 있다. 이것은 데이터 집합이 충분할 때, |P|는 $|I|^{maxL}$만큼 생성됨으로 maxL에 따라 급격히 증가한다. |U|는 어느 정도가 길이가 지나면 증가하지 않고 적절한 상수값에 급속히 가까워지는 것을 볼 수 있다. 그 이유는 같은 규칙이 다시 나올 확률은 적어지기 때문이다.

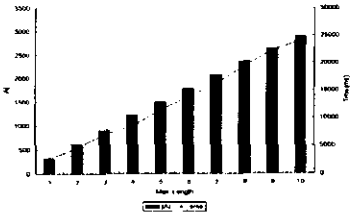
3. 3 maxD에 따른 |U|와 |P|

그림 5를 보면 |P|와 |U|가 같은 크기로 커져 가는 것을 볼 수 있다. 만약 maxD가 계속 커진다고 했을 때에도 |U| > |P|가 됨을 알 수 있다. maxL과 많은 차이를 보이는 이유는 maxL이 |P|를 $|I|^{maxL}$만큼 증가시키는데 비해, maxD는 maxD배 만큼 증가시키기 때문이다.



[그림 5] maxD에 따른 |P|와 |U|

3. 4 수행시간 측정



[그림 6] |A|와 수행 시간

수행시간에 있어서 |P|와 |U|의 비율보다는 |A|가 좌우한다. |A|는 데이터베이스에 접근하는 회수이기 때문에 |A|를 줄이는 것이 전체적인 성능을 높이는데 중요하게 작용될것으로 보인다.

4. 결론 및 향후 연구

본 논문에서는 데이터베이스와 강 결합됨으로써 표준화에 대한 문제를 해결하고자 시도하였다. 데이터베이스와의 강 결합은 표준화 문제를 해결함과 더불어 마이닝 시스템에 DBMS의 관련 기술을 이용함으로써 성능 향상을 시도하였다.

본 논문에서는 거리와 길이의 개념을 연관규칙에 추가 확장하였다. 또한 연관규칙의 트랜잭션 기반 데이터베이스의 적용 한계를 시계열 데이터에 적절하도록 변환하여 알고리즘을 설계하였다.

효율을 높이기 위해서 순차패턴탐사에서와 같이 빈발규칙을 찾아내는 단계의 추가가 필요하다.

[참고문헌]

[1]. H. Mannila, "Data mining: machine learning, statistics, and databases", Eight International Conference on Scientific and Statistical Database Management, Stockholm June 18-20, p. 1-8, 1996.

[2]. R. Agrawal and R. Srikant, "Mining Sequential Patterns", Proceedings of the 11th Data Engineering, 1995.

[3]. I. H. Witten and E. Frank, *Data Mining pratical machine learning tools and techniques with java implementations*, Morgan kaufmann publishers, 2000.

[4]. 김남호, 이동하, 이제현, 이전영, "연관규칙 탐사를 이용한 웹 사용자 패턴분석 기법", 정보처리학회 97추계 학술발표논문집, 제4권 2호, 425-430, 정보처리학회, 1997.

[5]. P. Adriaans and D. Zantinge, *Data Mining*, Addison Wesley Longman, 1996.

[6]. J. Han, "Database and data mining coupling", Panel discussion presentation at KDD'98 for the panel "Database-Data Mining Coupling", New York, USA, Aug. 1998.