한시적 연관규칙을 위한 데이타 주도 탐사 기법

조 일 래[†] · 김 종 덕[††] · 이 도 헌[†††]

요 약

연관규칙(association rule) 탐사(mining)는 대용량 데이타베이스로부터 사건간의 동시 발생 경향을 발견하는 작업이다. 기존의 연관규칙은 전체 트랜잭션에 대하여 성립하는 사건간의 연관 관계만을 고려하고 있다. 그러나 어떤 연관규칙은 비록 전체 시간구간에 대해서는 신뢰도가 그리 높지 않더라도 특정 기간에서 특별히 강한 신뢰도로 성립할 수 있고, 이러한 정보를 알 수 있다면 의사 결정에 매우 유용하리라고 생각한다. 본 논문에서는 임의의 부분 시간구간에서 특별히 높은 신뢰도를 갖는 연관성을 한시적 연관규칙(transient association rule)이라 정의하고, 대용량의 데이타베이스로부터 한시적 연관규칙이 성립하는 시간구간을 탐사하는 효율적인 알고리즘을 제안한다. 제안된 알고리즘은 불필요한 구간 검색을 배제할 수 있는 데이타 주도(data-driven) 검색 기법을 제시하고, 한 번의 데이타베이스 스캐닝(scanning)으로 다음 단계의 검색에 필요한 정보를 획득하여 주기억장치 상에 관리할 수 있도록 하는 효과적인 자료구조를 설계한다. 아울러 실험을 통해, 제안 알고리즘이 현장에 적용할 만한 시간 비용으로 수행됨을 보인다.

Data-Driven Exploration for Transient Association Rules

Il Rae Cho[†] · Jong Deok Kim[††] · Do Heon Lee[†††]

ABSTRACT

The mining of association rules discovers the tendency of events occurring simultaneously in large databases. Previously announced research on association rules deals with associations with respect to the whole transaction. However, some association rules could have very high confidence in a sub-range of the time domain, even though they do not have quite high confidence in the whole time domain. Such kind of association rules are expected to be very useful in various decision making problems. In this paper, we define transient association rule, as an association with high confidence worthy of special attention in a partial time interval, and propose an efficient algorithm which finds out the time intervals appropriate to transient association rules from large databases. We propose the data-driven retrieval method excluding unnecessary interval search, and design an effective data structure manageable in main memory obtained by one scanning of database, which offers the necessary information to next retrieval phase. In addition, our simulation shows that the suggested algorithm has reliable performance at the time cost acceptable in application areas.

※이 논문은 1996년도 전남대학교 학술 연구비 지원에 의하여 연구되었음.

† 정 회 원:순천공업전문대학 전자계산과 조교수
†† 정 회 원:전남대학교 대학원 전산통계학과 박사과정
††† 정 회 원:전남대학교 전산학과 전임강사

논문접수:1997년 1월 27일, 심사완료:1997년 4월 2일

1. 서 론

최근 여러 응용 분야에서 데이타베이스의 활용이 급증하고, 각 분야별로 데이타베이스에 저장하는 데이타의 양이 급속히 증가됨에 따라, 대용량 데이타베이스로부터 유용한 지식을 발견하고자 하는 데이타마이닝(data mining) 기술에 대한 연구가 활발히 이루어지고 있다. 데이타마이닝이란 데이타베이스 내에 기록된 원시 데이타로부터 단순 통계학적인 처리로는 유추가 불가능하지만, 잠재적(potentially)으로 유용한 새로운 형태의 정보들을 발견하는 작업을 의미한다[1]. 발견할 수 있는 지식의 종류는 응용 분야의 요구에 따라 달라지며, 데이타 요약(summary), 분류(classification), 순차 유형(sequential pattern), 집단화(clustering), 그리고 연관규칙(association rule) 등이 연구되고 있다[2, 3].

이들 중에서 특히 연관규칙은 서로 다른 사건간의 동시 발생 경향을 표현한 것으로서, POS (Point-Of-Sale) 데이타베이스에서의 구매패턴 분석[5], 전자교환시스템의 모듈 고장분석[14] 등 다양한 응용분야에서 실용성이 입증되어, 최근 가장 활발히 연구되고 있는 분야 중 하나이다. POS 데이타베이스의 예를 든다면, "냉장고를 구입한 고객은 에어컨을 함께 구입하는 경향이 있다."와 같은 연관규칙, 즉, 냉장고 구입이라는 사건과 에어컨 구입 사건간의 동시 발생 경향을 발견할 수 있다. 전자 교환 시스템의 고장 경고 메시지를 저장한 데이타베이스에서도 특정 모듈의 고장 경고와 다른 모듈의 고장 경고가 주로 함께 발생하는 경향을 연관규칙으로 표현할 수 있다. 이렇게 발견한 지식을 다음 분기를 위한 마케팅 전략에 활용하거나, 시스템의 신뢰도 분석을 위한 기초 자료로 활용할 수 있게 된다.

주어진 데이타베이스에 나타날 수 있는 사건의 가짓수는 일반적으로 매우 많다. 예를 들어, POS 데이타베이스에서는 해당 소매업체에서 판매하고 있는 모든 물품 가짓수가 바로 사건의 가짓수가 되고, 전자 교환 시스템의 경우에는 각 모듈에서 발생할 수 있는 경고 메시지의 총합이 사건의 가짓수가 된다. 그런데 연관규칙은 이러한 사건들의 부분집합으로 표현되므로 연관규칙의 가짓수는 사건의 가짓수에 대하여 지수적으로 증가하게 된다. 따라서 기존의 연관규칙 탐사 연구는 검색 공간을 효과적으로 줄여서 실용적인 시간 비용 내에 모든 유용한 연관규칙을 찾고자하는 노력에 주로 집중되어 왔다[5, 6, 7, 8, 15, 16, 17]. 또한, 일반화(generalization) 개념을 고려한 다단계 연관규칙의 탐사[9, 10]나, 범주(category)와 수량(quantitative) 속성을 포함하는 연관규칙의 연구[13]도 진행되고 있다.

기존의 연관규칙 탐사는 전체 데이타베이스에 대하여 연관규칙을 발견하기 때문에, 비록 전체 시간구간에서는 상대적으로 크게 드러나지 않지만, 부분 시간구간에서는 확연히 드러나는 연관규칙을 놓칠 수 있다는 문제점을 갖는다. 예를 들어, "가죽외투를 사면 스웨터를 함께 산다."와 같은 경향은 사계절 전체로 본다면 그 판매량의 상대적인 비율이 적어서 연관규칙으로서 강하게 드러나지 않더라도 겨울철에 판매된 데이타에 대해서만 탐사를 진행하면 명확히 나타날 수도 있다. 따라서, 본 연구에서는 연관규칙의 한 변형으로서 주어진 데이타베이스에 대해 부분 시간구간에서 강하게 성립하는 연관규칙을 한시적(transient) 연관규칙이라 정의하고, 데이타 주도(datadriven) 방식을 사용하여 한시적 연관규칙이 성립되는 부분 시간구간을 효율적으로 발견하는 한시적 연관규칙 탐사 알고리즘(TARMA:Transient Association Rules Mining Algorithm)을 제안한다.

한시적 연관규칙은 기존의 연관규칙에 사건 발생 시간구간을 추가한 것으로 이해될 수 있다. 그러나, 부분 시간구간에 강하게 성립하는 연관규칙을 탐사하는 문제는 시간 경계가 미리 주어지지 않을 경우 간단히 해결할 수 있는 문제가 아니다. 위의 예에서 본다면 월별 혹은 계절별로 기간을 미리 분할하고 특정 연관규칙이 어느 기간에 가장 높은 신뢰도를 갖게 되는지 평가한다면 매우 간단한 문제이다. 하지만, 임의의 기간, 예를 들어 7월 12일부터 8월 5일까지 특별히 높은 신뢰도를 가질 수 있다는 가능성을 놓치지 말아야 한다는 상황이라면 문제는 매우 복잡해진다. 간단한 계산으로 복잡도를 예시해 보자. 지난 5년간 발생한 사건들에 대하여 일별로 레코드가 저장되어 있고 임의의 기간을 고려한다면, 가능한 기간의 가짓수는 (5년*365일)*(5년*365일−1)/2=1,660,750개가 된다. 이것은 A* 검색 기법[18]을 사용했을 때 검색 트리의 노드 개수가 무려 166만개가 된다는 것을 의미

한다. 이와 같은 임의 구간 설정의 문제 복잡도는 수량 연관규칙에 대한 연구[13]에도 존재하는데, 아직 완전한 해결 방안은 제시되지 않고 있다.

본 연구에서는 이와 같은 복잡도를 해결하여 대용량의 데이타베이스에 실제 적용 가능한 한시적 연관규칙 탐사 기법을 제안하고, 그 결과를 실험을 통해 검증하고자 한다. 제안된 TARMA는 전체 데이타베이스에 대해 최소한의 일정 비율 이상을 만족하는 연관규칙을 대상으로 부분 시간구간 탐사를 진행하기 위해, 기존의 연관규칙 탐사 알고리즘에 상대적으로 낮은 임계치(threshold)를 적용한 수행 결과가 존재함을 가정하고, 구해진 각 연관규칙의 신뢰도가 특별히 높게 나타나는 시간구간을 도출한다. 특히 알고리즘 수행 성능을 향상시키기 위해 불필요한 검색을 배제할 수 있는 데이타주도(data-driven) 최적화 검색 기법을 제시하고, 데이타베이스 스캐닝(scanning) 회수를 최소화하기 위하여 한 번의 데이타베이스 스캐닝으로 다음 단계의 검색에 필요한 정보를 획득하여 주기억장치 상에 관리할 수 있도록 하는 효과적인 자료구조를 설계한다.

본 논문의 구성은 다음과 같다. 2절은 한시적 연관규칙을 정의하고, 그 타당성 척도를 제시하며, 데이타주도 방식의 데이타베이스 분할 기준을 서술한다. 주어진 연관규칙이 강하게 성립하는 시간구간을 발견하는 TARMA 알고리즘을 3절에서 제안하고, 실험을 통한 TARMA의 성능 분석을 4절에서 소개한다. 5절은 본 논문의 주요 사항을 정리하고 추후 연구 방향을 언급한다.

2. 한시적 연관규칙

이 절에서는 기존의 연관규칙에 사건 발생 시간구간을 추가한 한시적 연관규칙을 정의하고, 그 타당성 척도를 제시한다. 아울러 시간구간 발견을 위한 데이타 주도 분할 방식이 제안된다. 본 논문에서는 설명의 편의를 위해 백화점이나 대형 유통센터에서 발생하는 고객의 구매 행위를 예로 들어 설명하지만, 일반적인 다른 응용 분야에도 적용 가능하다.

2.1 한시적 연관규칙의 정의

데이타베이스는 트랜잭션 집합 $T=\{t_1, t_2,..., t_n\}$로 나타낼 수 있다. 트랜잭션 t_i는 $\langle \text{tid}_i, \text{itemset}_i, \text{ts}_i \rangle$로 구성되는데, 한 고객이 한 번에 구매한 품목의 집합을 의미한다. 여기서 'tid_i'는 각 트랜잭션을 유일하게 구별하는 식별자이다. 집합 $I=\{i_1, i_2,...,i_m\}$를 고객의 구매 대상 품목 등을 나타내는 전체 항목집합(itemset)이라 할 때, 'itemset_i'는 항목집합 I의 부분 집합으로서, t_i가 표현하는 구매 행위에서 구매된 항목의 집합이다. 한편, 'ts_i'는 트랜잭션 t_i가 데이타베이스에 기록된 시간(timestamp)을 의미하며, 표현되는 시간의 단위는 편의상 일(day)로 가정하였으나 응용 분야와 사용자 요구에 따라 달라질 수 있다. 그리고, 데이타베이스에 포함된 트랜잭션들이 기록된 시간들의 집합을 $TIME=\{ts_1, ts_2,..., ts_n\}$이라 할 때, 첫 번째 트랜잭션과 마지막 트랜잭션이 기록된 시간을 시작점과 끝점으로 하는 시간구간(TI: Time Interval) (ts_1, ts_n)을 전체 시간구간 U라고 한다.

[정의 1] 부분 시간 구간에서 성립되는 연관규칙은 다음과 같은 형태로 정의된다.

한시적(transient) 연관규칙

$X \Rightarrow_T Y@[TI]$

단, X, $Y \subset I$, $X \cap Y = \phi$이고, X, Y를 각각 전제부(antecedent)와 결론부(consequent)라 한다. 그리고, $TI=\{(ts_i, ts_j),..., (ts_k, ts_l)\}$는 한시적 연관규칙이 성립하는 부분 시간구간의 집합을 나타내며, $ts_i, ts_j, ts_k, ts_l \in TIME$, $TI \subseteq U$이다.

비록 전체 시간구간에서는 상대적으로 크게 드러나지 않지만, 부분 시간구간에서는 확연히 드러나는 연관규칙을 한시적(transient) 연관규칙이라 하며, $X \Rightarrow_T Y@[TI]$는 '부분 시간구간 집합 TI에서 X를 구매하면 Y도 함께 구매하는 경향이 있다.'로 해석된다.

[예제 1] 한시적 연관규칙 '냉장고, 세탁기 $\Rightarrow_T$ 에어컨@[(96.6.1, 96.6.25), (96.7.15, 96.8.5)]'은 1996년 6월 1일부터 1996년 6월 25일 까지와 1996년 7월 15일부터 1996년 8월 5일 까지의 부분 시간구간 동안에 냉장고와 세탁기를 구입한 고객은 에어컨을 함께 구매하는 경향이 강하게 존재함을 나타낸다.

2.2 한시적 연관규칙의 타당성 척도

연관규칙의 타당성은 통계적 중요도(statis-tical signifi-

cant)와 규칙 강도(rule strength)의 두 가지 측면에서 평가된다[5]. 통계적인 중요도는 지지도(SD:Support Degree)로서 평가되는데, 전체 트랜잭션 중에서 해당 연관규칙의 전제부와 결론부를 포함하는 트랜잭션의 비율로 계산된다. 그리고, 규칙의 강도는 신뢰도(CF: Confidence Factor)로서 평가되는데, 규칙의 전제부를 포함하는 트랜잭션 중에서 결론부까지 함께 포함하는 트랜잭션의 비율로서 정량화된다. 기존의 연관규칙 탐사 과정은 전체 시간구간을 대상으로 사용자에 의해 주어지는 임계치(threshold value)인 최소지지도(*MinSup*:*Minimum Sup*port degree) 이상을 만족하는 다량(frequent) 항목집합을 발견하는 과정과, 발견된 다량 항목집합 내에 포함된 항목들 중에서 최소신뢰도(*MinConf*:*Min*imum *Conf*idence factor) 이상을 만족하는 항목들 간의 연관규칙을 생성하는 두 단계로 구성된다. 본 논문에서는, 부분 시간구간에 발생한 트랜잭션들을 대상으로 위에서 언급된 두 가지 척도를 적용한다.

[정의 2] 한시적 연관규칙 $X \Rightarrow_T Y@[TI]$의 타당성 척도와 성립 요건은 다음과 같이 정의된다.

- 임의의 부분 시간구간 $(ts_i, ts_j) \in TI$에서 항목집합 $X \subseteq I$의 지지도는 $SD(X) = |X|/k$이다. 단, $|X|$는 (ts_i, ts_j)에 발생된 트랜잭션들 중에 항목집합 X를 포함하는 트랜잭션 수이고, k는 (ts_i, ts_j)에 발생된 전체 트랜잭션의 수이다.
- 부분 시간구간 (ts_i, ts_j)에서 $X \Rightarrow_T Y@[TI]$의 신뢰도 CF는 $SD(X \cup Y)/SD(X)$이다.
- 만일 모든 $(ts_i, ts_j) \in TI$에서 $SD(X \cup Y) \geq MinSup$이고, $SD(X \cup Y)/SD(X) \geq MinConf$이면, 한시적 연관규칙 $X \Rightarrow_T Y@[TI]$가 성립한다고 정의한다.

간단한 구매 데이타베이스의 예와 항목별 지지도가 〈표 1〉에 제시되었고, 사용자에 의해 주어지는 임계치인 최소지지도를 60%, 최소신뢰도를 70%로 가정한다. 기존의 연관규칙 탐사는 전체 트랜잭션을 대상으로 각 항목집합의 지지도를 계산하므로, 최소지지도 이상을 만족하는 항목집합 {(1), (3), (1,3)}이 다량 항목집합으로 결정된다. 이들 다량 항목간의 신뢰도를 계산하여, 70% 이상의 신뢰도를 만족하는 연관규칙들이 다음과 같이 구해진다.

```
1⇒3 /* 지지도:60%, 신뢰도:75% */
3⇒1 /* 지지도:60%, 신뢰도:100% */
```

항목집합 (1,4)는 전체 트랜잭션에 대한 지지도가 40%이므로 주어진 최소지지도(60%)를 만족하지 못하여 연관규칙으로 생성되지 않는다. 그러나, 연관규칙 탐사에 상대적으로 낮은 임계치, 예를 들어, 최소지지도 30%와 최소 신뢰도 50%를 적용하면 항목집합 (1,4)도 다량 항목집합에 포함되고 다음과 같은 연관규칙이 성립한다.

```
1⇒4 /* 지지도:40%, 신뢰도:50% */
4⇒1 /* 지지도:40%, 신뢰도:100% */
```

연관규칙 "1⇒4"는 상대적으로 낮은 임계치를 적용하여 구해진 규칙이지만, 시간구간을 고려하여 데이타베이스를 분할하고, 각 분할 내에서 각 연관규칙의 지지도와 신뢰도를 구하면 예상치 못한 다른 결과를 얻을 수 있다. 예를 들어, 시간구간 (96.10.19, 96.10.23)에 발생한 두 개의 트랜잭션만을 대상으로 하면 연관규칙 "1⇒4"의 지지도는 100%가 되어 전체 시간구간에서의 지지도(40%)와는 매우 큰 차이를 나타낸다. 또한, 연관규칙 "1⇒3"도 부분 시간구간(96.10.13, 96.10.23)에서는 75%의 지지도를 갖고, (96.10.13, 96.10.17)에서는 100%의 지지도를 갖는다. 이러한 점에 착안하여, 본 논문에서는 상대적으로 낮은 임계치가 적용되어 얻은 연관규칙들을 대상으로 각 연관규칙이 강하게 성립하는, 즉, 상대적으로 높은 임계치를 만족하는 시간구간 정보를 기존의 연관규칙에 부가한 한시적 연관규칙의 발견을 목표로 하며, 이러한 정보는 계절별 판매 전략 수립 등의 의사 결정에 유용하리라고 생각한다. 〈표 1〉에서 구할 수 있는 한시적 연관규칙의 예는 다음과 같다.

```
1⇒T 4 @[(96.10.19, 96.10.23)]
      /* 지지도:100%, 신뢰도:100% */
4 ⇒T 1 @[(96.10.19, 96.10.23)]
      /* 지지도:100%, 신뢰도:100% */
1 ⇒T 3 @[(96.10.13, 96.10.23)]
      /* 지지도:75%, 신뢰도:75% */
3 ⇒T 1 @[(96.10.13, 96.10.23)]
      /* 지지도:75%, 신뢰도:100% */
```

〈표 1〉 데이타베이스의 예와 항목별 지지도
〈Table 1〉 An example database and the support degree of each itemset

tid	*itemset*	*timestamp*
100	1, 2, 3	96. 10. 13
200	1, 3	96. 10. 17
300	1, 4	96. 10. 19
400	1, 3, 4	96. 10. 23
500	2, 5, 6	96. 10. 25

항목 번호	각 항목을 포함하는 트랜잭션 수	지지도 (%)
1	4	80
2	2	40
3	3	60
1,2	1	20
1,3	3	60
1,4	2	40
1,2,3	1	20
...	...	...

2.3 데이타 주도 분할

한시적 연관규칙이 성립되는 시간구간을 발견하기 위해서는 데이타베이스를 일정 기준에 의해 분할(partition)하고, 각 분할에 대하여 규칙의 성립 여부를 확인해야 한다. 그런데, 주어진 데이타베이스의 전체 시간구간을 인위적인 경계, 예를 들어 월, 계절, 분기 등으로 나누지 않고 임의의 시간구간으로 나눌 수 있는 가짓수는 매우 많기 때문에 효율적인 분할 방법이 요구된다. 이 절에서는, 임의의 연관규칙에 대하여 불필요한 시간구간을 고려하지 않도록 하는 데이타 주도(data-driven) 분할 방식을 제시한다.

데이타베이스에 포함된 각 트랜잭션을 주어진 연관규칙과의 관계에 따라 다음과 같이 분류한다[19]. 주어진 연관규칙의 전제부를 포함하는 트랜잭션을 관련 실례(relevant instance)라 하고, 그렇지 않은 트랜잭션을 무관 실례(irrelevant instance)라고 한다. 관련 실례를 다시 세분하여, 결론부까지를 포함하는 긍정 실례(positive instance)와 결론부를 포함하지 않는 부정 실례(negative instance)로 구분한다. 긍정 실례는 연관규칙의 지지도와 신뢰도를 모두 증가시키는 요소이며, 부정 실례는 전제부만을 포함하므로 규칙의 강도인 신뢰도를 저하시키는 요소이다. 또한, 무관 실례는 지지도를 저하시키는 요소이므로, 모든 유형의 트랜잭션 수효가 반드시 파악되어야 한다. 예를 들어, 〈표 1〉에서 성립하는 연관규칙 '1⇒3'에 대하여, tid가 100인 트랜잭션은 항목 '1'과 '3'을 모두 포함하는 긍정 실례이고, tid가 300인 트랜잭션은 부정 실례, 그리고 tid가 500인 트랜잭션은 무관 실례이다.

임의의 데이타베이스 내에 n개의 트랜잭션이 존재한다면, 연속된 트랜잭션에 의해 생성될 수 있는 분할의 가짓수, 즉 시간구간의 수는 $n(n-1)/2$이다. 그러나, 서로 다른 긍정 실례를 각 분할의 시작점과 끝점으로 이용하는 데이타 주도 방식을 이용하면, 긍정 실례의 수를 k라 할 때, 분할의 가짓수는 $k(k-1)/2$가 된다. 그런데, 긍정 실례의 수는 지지도에 따라 다르지만 대부분 전체 트랜잭션의 수에 비해 월등히 적다. 즉, $k \ll n$이므로, 생성되는 분할의 수를 크게 줄일 수 있다. 따라서, 제안된 데이타 주도 방식은 무관 실례들만으로 구성되는 분할 등 고려할 필요가 없는 시간구간의 생성을 방지한다.

[정의 3] 임의의 연관규칙에 대한 서로 다른 두 긍정 실례 $t_i, t_j \in T$에 의해 형성된 시간구간 (ts_i, ts_j) 내의 기록 시간(timestamp) 값을 갖는 트랜잭션 집합을 분할(partition) $\rho_{i,j}$라 정의하고, 서로 다른 분할 $\rho_{a,b}$와 $\rho_{c,d}$는 중첩될 수 있다.

만일, 중첩된 분할들을 고려하지 않으면 분할이 이루어진 경계선에서 시간 정보의 상실(loss)이 발생한다. 예를 들어, 시간의 최소 단위를 일(day)로 가정했을 때, 연속된 시간구간 '1개월'은 매월 1일부터 말일까지 뿐만 아니라, 2일부터 다음 달 1일까지 등 30여 개의 가짓수가 존재한다. 따라서, 서로 중첩되지 않는 분할에 대하여 규칙의 존재 여부를 확인하여 결론지으면 많은 다른 시간구간에서 성립될 수 있는 한시적 연관규칙의 발견이 불가능하므로 중첩된 분할이 반드시 요구된다.

우리는 탐사 결과로서 부분 시간구간이 부가된 한시적 연관규칙을 얻는다. 그러나, 너무 적은 수의 트랜잭션 내에서 성립되는 한시적 연관규칙이나 시간구간의 크기가 너무 작은 규칙, 예를 들어, 전체 트랜

잭션 수를 100만개, 전체 시간구간을 5년으로 가정할 때, 단지 10개의 트랜잭션에서 성립되는 규칙이나, 3일간의 시간구간에서 성립되는 규칙 등을 제외할 수 있도록 [정의 4]와 같은 타당성 척도를 부가하며, 제안되는 두가지 척도 값은 응용 분야의 특성에 따라 사용자에 의해 결정된다.

[정의 4] 임의의 분할이 포함하는 긍정 실례의 최소 수는 최소 분할계수(minimum partition counter) Ψ로 정의하고, 한시적 연관규칙이 성립되는 시간구간의 최소 크기는 최소 시간구간(minimum time interval) Ω라 정의한다.

3. 한시적 연관규칙 탐사 알고리즘

본 절에서 제안하는 한시적 연관규칙 탐사 알고리즘(TARMA : Transient Association Rule Mining Algorithm)은 상대적으로 낮은 임계치가 적용된 기존 연관규칙 탐사 알고리즘의 수행 결과로 발견된 연관규칙들이 존재함을 가정하고, 각 연관규칙이 강하게 성립하는 부분 시간구간을 발견한다. 이 절에서는 먼저 단일 연관규칙에 대한 부분 시간구간 탐사 알고리즘인 TARMA를 예제와 함께 소개하고, 복수개의 연관규칙을 수용하는 확장_TARMA를 제시한다.

3.1 단일 연관규칙에 대한 부분 시간구간 탐사

임의의 단일 연관규칙에 대한 부분 시간구간 탐사하는 TARMA는 먼저 해당 규칙의 긍정 실례를 기준으로 분할 기준 테이블(partition base table)을 구성한다. 분할 기준 테이블의 엔트리(entry)는 데이타베이스 분할에 이용되고, 각 엔트리를 점진적으로 합병하여 한시적 연관규칙이 성립되는 시간구간을 발견한다.

(그림 1)에 제시된 TARMA는 다음과 같이 3 단계로 구성되며, 데이타베이스 내의 트랜잭션들은 기록된 시간 값을 기준으로 정렬되었음을 가정한다.

[단계 1] 주어진 연관규칙의 긍정실례를 추출하여 분할 기준 테이블(partition base table)을 구성한다.

[단계 2] [단계 1]의 결과와 최소 분할 계수 Ψ를 이용하여 첫 번째 분할을 형성하고, 최소 시간구간 Ω보다 큰 시간구간을 구한다.

[단계 3] [단계 2]의 모든 분할에 대하여 분할의 마지막 긍정실례에 인접한 다음 긍정실례를 점진적으로 합병하여 각 분할을 확장하면서 시간구간 값을 구한다.

```
Algorithm TARMA(D,ri)
입력: 데이타베이스 D, 연관규칙 ri, MinSup,
MinConf, 최소 분할 계수 Ψ, 최소 시간구간 Ω
출력 : 한시적 연관규칙 ri@[TI]
TI = ∅;
ri에 대한 분할 기준 테이블 B 생성;
                              /* 단계 1 */
B와 최소 분할 계수 Ψ에 의해 T[1] 생성;
                              /* 단계 2 */
 forall 분할 ρi,j ∈ T[1] do begin
    TI =TI∪{규칙이 성립되는 시간구간 중 Ω
          보다 큰 값};
 end
 for k=2 to (긍정실례수-Ψ+1) do begin
                              /* 단계 3 */
    B와 T[k-1]을 이용하여 T[k] 생성;
    forall 분할 ρi,j ∈ T[k] do begin
       TI =TI∪{규칙이 성립되는 시간구간 중
             Ω보다 큰 값};
          TI에서 중복되는 시간구간 제거;
    end
 end
 END_simple_TARMA
```

(그림 1) TARMA 알고리즘
(Fig. 1) TARMA algorithm

제안된 TARMA의 [단계 3]에서 합병은 "긍정실례수−Ψ(최소분할계수)+1"회 만큼 수행되며, Ψ(최소분할계수)가 2인 경우는 (긍정실례수−1)이 합병 회수가 된다. 즉, 각 긍정실례로부터 최소분할계수만큼의 긍정실례를 포함하는 모든 분할에 대하여 인접한 다음 긍정실례를 포함하는 확장된 분할로 합병이 계속되는데, 이는 탐사 과정에서 발생할 수 있는 발견하지 못하는 부분 시간구간의 발생을 방지한다. 특정 연관규칙에 대한 긍정 실례의 분포는 전혀 예상할 수 없으므로 일정 시간구간에서 집중적으로 발생된 긍정실례로 인하여 최소지지도를 만족하는 보다 긴 시간구간이 발견될 가능성은 항상 존재하고, 이를 위하

여 마지막 긍정실례까지의 반복 확장이 불가피하다.

3.1.1 분할 기준 테이블 생성

임의 크기의 시간구간에서 성립되는 한시적 연관규칙을 발견하기 위해서는 가장 작은 기간부터 순차적으로 점점 기간을 합병해 가는 상향식 방식을 이용해야 한다. 그러나, 이러한 과정을 디스크주도 데이타베이스에 그대로 적용한다면 모든 인접한 기간의 쌍들을 순차적으로 병합해야 하므로, 데이타베이스 전체 스캐닝 회수가 전체 레코드의 개수에 비례하여 증가하게 된다.

이 절에서는 데이타베이스 스캐닝 회수를 최소화하기 위하여 한 번의 데이타베이스 스캐닝으로 다음 단계의 검색에 필요한 정보를 획득하여 주기억장치상에 관리할 수 있도록 하는 효과적인 자료구조인 분할 기준 테이블을 제안한다. 긍정 실례를 기초로 구성되는 분할 기준 테이블 B의 각 엔트리는 〈tid, n_num, i_num〉로 이루어지며, 각 요소의 의미는 다음과 같다.

- tid는 주어진 연관규칙에 대한 긍정 실례의 트랜잭션 번호이다.
- n_num, i_num은 각각 바로 이전(previous) 긍정 실례부터 현(current) 긍정 실례 사이에 존재하는 부정 실례와 무관 실례의 수이다.

분할 기준 테이블은 데이타베이스 내의 긍정 실례 수만큼의 엔트리를 갖으며, 각 엔트리는 데이타베이스를 분할하는 시점과 종점이 되고, 각 분할의 확장시 반복되는 과정에 분할 내의 지지도와 신뢰도 계산에 필요한 기본 자료를 제공한다. 예를 들어, 〈표 2〉와 같은 구매 데이타베이스 에에서 성립되는 단일 연관규칙 '1⇒3'에 대한 분할 기준 테이블은 〈표 3〉과 같다.

분할 기준 테이블 B의 tid가 '300'인 두 번째 엔트리는 바로 이전 긍정 실례인 '100'번 트랜잭션과 현 긍정 실례인 '300'번 사이에 하나의 부정실례(200번 트랜잭션)만이 존재하므로 n_num은 '1'이 되고, i_num은 '0'이 된다. 또한, tid가 '700'인 세 번째 엔트리는 이전 긍정 실례인 '300'번 트랜잭션과 자신의 사이에 두 개의 부정실례(500, 600번 트랜잭션)와 하나의 무관실례(400번 트랜잭션)가 존재하므로 n_num은 '2'가 되고, i_num은 '1'이 된다.

〈표 2〉 구매 데이타베이스의 예
〈Table 2〉 Example of purchase database

tid	*itemset*	*timestamp*
100	1 2 3	96. 9. 1
200	1 4	96. 9. 7
300	1 3 5	96. 9. 9
400	2 5 6	96. 9. 13
500	1 4 8 9	96. 9. 13
600	1 4 9	96. 9. 15
700	1 3 5	96. 9. 16
800	3 5	96. 9. 17
810	3 1 7 8	96. 9. 19
820	1 3	96. 9. 20
830	1 3 5 8	96. 9. 23
840	4	96. 9. 23
900	1 3 4	96. 9. 27
...	...	...

〈표 3〉 분할 기준 테이블 B
〈Table 3〉 Partition base table B

tid	*n_num*	*i_num*
100	0	0
300	1	0
700	2	1
810	0	1
820	0	0
830	0	0
900	0	1
...	...	...

3.1.2 분할 테이블 생성과 합병

이 단계에서는 분할 기준 테이블의 각 엔트리를 시점과 종점으로 하여 데이타베이스를 중첩 분할한다. 특정 연관규칙에 대한 최초 분할은 최소 분할 계수 Ψ 만큼의 긍정 실례를 포함하며, 분할 내에 존재하는 부정 실례와 무관 실례의 개수를 분할 기준 테이블 B로부터 구하여 첫 번째 분할 테이블 $T[1]$을 생성한다. 분할 테이블의 각 엔트리는 〈tid_1, tid_2, n_num, i_num〉로 구성된다. tid_1과 tid_2는 분할의 시점과 종점이 되는 트랜잭션 번호이고, n_num과 i_num은 tid_1과 tid_2 사

이에 존재하는 부정 실례와 무관 실례의 수이며, 다음과 같이 산출된다.

$T[1].tid_1^k = B.tid_k$
$T[1].tid_2^k = B.tid_{k+\Psi-1}$
$T[1].\text{n\_num}_k = B.\text{n\_num}_{k+1} + \ldots + B.\text{n\_num}_{k+\Psi-1}$
$T[1].\text{i\_num}_k = B.\text{i\_num}_{k+1} + \ldots + B.\text{i\_num}_{k+\Psi-1}$

단, k는 1부터 (긍정실례수$-\Psi+1$)까지 1씩 증가하는 정수 값으로 테이블의 k번째 엔트리를 의미하며, $T[1].tid_1^k$과 $T[1].tid_2^k$는 분할 테이블 $T[1]$의 k번째 엔트리의 tid_1과 tid_2를 나타낸다. 예를 들어, Ψ를 2로 가정하면, 〈표 3〉을 이용한 $T[1]$은 〈표 4〉와 같다.

〈표 4〉 1 단계 분할 테이블 $T[1]$
〈Table 4〉 1′st partition table $T[1]$

tid_1	tid_2	n_num	i_num
100	300	1	0
300	700	2	1
700	800	0	1
810	820	0	0
820	830	0	0
830	900	0	1
...	...	...	...

분할 테이블 $T[1]$의 k번째 분할 ρ_k에서 부분 항목집합 (1, 3)의 지지도 $SD_{T[1]}(\rho_k)$와 신뢰도 $CF_{T[1]}(\rho_k)$는 다음과 같이 구해진다.

$$SD_{T[1]}(\rho_k) = \frac{\text{긍정실례수}}{\text{분할내트랜잭션총수}} = \frac{2}{(T[1].\text{nnum}_k + T[1].\text{inum}_k + 2)}$$

$$CF_{T[1]}(\rho_k) = \frac{\text{긍정실례수}}{(\text{긍정실례수} + \text{부정실례수})} = \frac{2}{(2 + T[1].\text{nnum}_k)}$$

구해진 지지도와 신뢰도가 각각 *MinSup*, *MinConf* 이상이면, 분할 ρ_k는 한시적 연관규칙이 성립되는 구간이 되며, $T[1].tid_1^k$와 $T[1].tid_2^k$의 시간 값을 구매 데이타베이스에서 얻는다. 구해진 시간 값의 크기가 최소 시간구간 Ω보다 크면, 발견하고자 하는 최종 시간구간의 하나이므로 TI_i에 추가된다.

분할 테이블 $T[1]$에 의해 생성되는 분할은 최소 크기의 분할이며, 모든 분할은 분할의 종점이 되는 마지막 긍정 사례에 인접한 다음 긍정 사례를 추가하여 분할을 확장한다. 예를 들면, $T[1]$에서 첫 번째 분할은 (100, 300) 사이에 존재하는 트랜잭션들로 구성되며, 300번 트랜잭션에 인접한 다음 트랜잭션인 700번 트랜잭션을 포함하는 (100, 700)번 사이의 트랜잭션들이 두 번째 단계의 분할인 $T[2]$의 첫 번째 엔트리가 된다. 확장된 분할 테이블 $T[2]$는 분할 기준 테이블 B와 분할 테이블 $T[1]$을 이용하여 구성한다. 예를 들어, 〈표 3〉과 〈표 4〉를 이용하여 구성된 $T[2]$는 〈표 5〉와 같다.

〈표 5〉 2 단계 분할 테이블 $T[2]$
〈Table 5〉 2′nd partition table $T[2]$

tid_1	tid_2	n_num	i_num
100	700	3	1
300	810	2	2
700	820	0	1
810	830	0	0
820	900	0	1
...	...	...	...

이와 같이 반복되는 합병 과정을 일반화된 형태로 정형화하면 다음과 같다. j 번째 분할 테이블 $T[j]$는 분할 기준 테이블 B와 분할 테이블 $T[j-1]$을 이용하여 구해지며, 분할 테이블 $T[j]$의 k 번째 엔트리의 값은 다음과 같다.

$T[j].tid_1^k = T[j-1].tid_1^k$
$T[j].tid_2^k = T[j-1].tid_2^{k+1}$
$T[j].\text{n\_num}_k = T[j-1].\text{n\_num}_k + B.\text{n\_num}_{j+\Psi+k-2}$
$T[j].\text{i\_num}_k = T[j-1].\text{i\_num}_k + B.\text{i\_num}_{j+\Psi+k-2}$

분할 테이블 $T[j]$의 모든 분할은 동일하게 ($\Psi+j-1$)의 긍정 실례를 포함하며, k 번째 분할 ρ_k의 부분 항목집합 'X'와 'Y'에 대한 지지도 $SD_{T[j]}(\rho_k)$와 신뢰도 $CF_{T[j]}(\rho_k)$는 다음과 같이 구해진다.

$$CD_{T[j]}(\rho_k) = \frac{(\Psi + j - 1)}{(T[j].nnum_k + T[j].inum_k + \Psi + j - 1)}$$

$$CF_{T[j]}(\rho_k) = \frac{(\Psi + j - 1)}{(T[1].nnum_k + \Psi + j - 1)}$$

한시적 연관규칙의 조건을 만족하는 분할 ρ_k의 시간구간에서 Ω보다 큰 구간만을 TI_i에 추가한다. 또한, 구해진 시간구간이 이전 단계에서 구해진 시간구간을 포함하면, 이전 단계의 작은 시간구간을 제거하여 중복되는 결과의 발생을 방지한다. 예를 들면, $T[j-1]$까지의 과정에서 TI_i에 시간구간 ('96.4.1, '96.4.15)이 존재하고 $T[j]$에서 ('96.4.1, '96.4.20)이 얻어지면, 중복된 결과를 발생하므로 ('96.4.1, '96.4.15)를 제거하고 ('96.4.1, '96.4.20)만을 결과로 구한다.

3.2 복수 연관규칙에 대한 부분시간구간 탐사

TARMA에 입력되는 연관규칙이 복수개이면, 각각의 연관규칙에 대하여 3.1절에서 제시한 TARMA를 적용해야 하므로, 입력된 연관규칙 수가 커질 경우에 데이타베이스 스캐닝 횟수가 너무 많아질 수 있는 문제점이 발생한다. 그러나, 복수개의 연관규칙에 대해서도 한번의 데이타베이스 스캐닝으로 각 연관규칙의 연속된 다음 탐사에 이용할 수 있는 요약 정보의 추출이 가능하다. 분할 기준 테이블은 각각의 연관규칙에 대해 독립적으로 구성되며, 입력된 연관규칙의 수만큼의 분할 기준테이블이 만들어진다. (표 2)와 같은 구매 데이타베이스의 예에서 다음과 같은 세 개의 연관규칙이 전체 시간구간에서 성립된다고 가정하자. 이러한 세 개의 연관규칙에 대한 분할 기준테이블은 (그림 2)와 같다.

분할기준테이블들이 구성되는 과정은 단일 연관규칙에 대한 분할기준테이블을 구성하는 과정과 동일하다. 그러나, 하나의 트랜잭션이 하나 이상의 연관규칙에 대해 긍정실례가 될 수 있으므로 같은 tid가 두 개 이상의 분할기준테이블에 기록될 수 있다. 예를 들어, tid가 '300'인 트랜잭션은 연관규칙 r_1의 긍정실례(항목 '1'과 '3'을 모두 포함하므로)인 동시에 r_3의 긍정실례이므로 두 분할기준테이블에 기록되지만 각 엔트리의 n_num와 i_num은 서로 다른 값을 갖는다. 입력된 각 연관규칙들에 대한 시간구간 탐사는 자신의 분할기준테이블을 기준으로 3.1절에서 제시한

tid	n_num	i_num
100	0	0
300	1	0
700	2	1
810	0	1
820	0	0
830	0	0
900	0	1
...	...	...

(a) 연관규칙 r_1의 분할기준테이블 B[1]

tid	n_num	i_num
200	1	0
500	1	1
600	0	0
900	4	2
...	...	...

(b) 연관규칙 r_2의 분할기준테이블 B[2]

tid	n_num	i_num
300	0	2
700	1	2
830	1	2
...	...	...

(c) 연관규칙 r_3의 분할기준테이블 B[3]

(그림 2) 복수개의 연관규칙에 대한 분할기준테이블
(Fig. 2) Partition base tables of multiple association rules

TARMA의 [단계 2]와 [단계 3]을 반복 적용한다.

기존의 연관규칙 탐사 알고리즘의 수행 과정에 상대적으로 낮은 최소지지도와 최소신뢰도가 적용되어 구해진 복수개의 연관규칙들을 입력으로 받아들여, 각각의 연관규칙에 대한 시간구간을 탐사하는 확장_TARMA가 (그림 3)에 제시되었다. 알고리즘 기술 과정에서 $r_i.I_A$와 $r_i.I_C$는 각각 입력된 임의의 연관규칙 r_i의 전제부와 결론부에 해당하는 항목집합을 의미한다.

Algorithm Extended_TARMA
입력 : 데이타베이스 $D = \{t_1, t_2, ..., t_n\}$,
연관규칙 집합 $R = \{r_1, r_2, ..., r_k\}$,
MinSup, *MinConf*,
최소 분할 계수 Ψ, 최소 시간구간 Ω
출력 : 한시적 연관규칙 집합 $\{r_i.@[TI_i]\}$, i=1,k

```
Read_in_conventional_association_rule_set;

for j = 1 to k do begin
      /* 각 연관규칙에 대한 PBT 구성 */
   rj.n_num = 0;
```

```
    rj.i_num = 0;
    cnt(j) = 1;
  end
  for i = 1 to n do begin
    for j = 1 to k do begin
      if (rj.IA ⊂ ti.itemset) then
          if (rj.IC ⊂ ti.itemset) then begin
              B[j].tid_cnt(j) = ti.tid;
              B[j].n_num_cnt(j) = rj.n_num;
              B[j].i_num_cnt(j) = rj.i_num;
              cnt(j)=cnt(j)+1;
              rj.n_num = 0;
              rj.i_num = 0;
              end
          else
              rj.n_num = rj.n_num + 1;
        endif
      else
          rj.i_num = rj.i_num + 1;
      endif
    end
  end
for i = 1 to k do begin
        /*TARMA의 단계2, 3 반복 수행*/
 각 연관규칙에 대한 분할테이블 생성과 합병;
end
END_Extended_TARMA
```

(그림 3) 확장_TARMA 알고리즘
(Fig. 3) Extended_TARMA Algorithm

4. 성능 분석

이 절에서는 합성(synthetic) 데이타를 이용한 모의 실험을 통해 TARMA 알고리즘의 성능을 분석한다. 실험에 사용된 합성 데이타는 [6]에서 제시된 무작위(randomized) 방식을 이용하여, 구매 현장에서 발생될 수 있는 실 데이타의 특성이 반영된 데이타로서 3.1.1절에서 제시된 〈표 2〉와 같은 유형이다. 알고리즘의 구현은 SunOS 5.5를 운영체제로 하고, 128MB의 주기억 용량과 CPU 클럭(clock) 속도 167MHz을 가진 AXIL Ultra-SPARC 워크스테이션 상에서 이루어졌다.

지금까지 진행되어온 연관규칙 탐사를 위한 알고리즘의 성능 분석은 알고리즘 수행 과정에 형성되는 다량 항목집합에 대한 후보(candidate) 항목집합[5]의 감소 비율과, 최소지지도의 변화에 따라서 알고리즘의 수행 시간이 안정된 변화를 가짐을 보이고 있다 [5, 6, 8, 15, 16]. 일반적으로, 최소지지도가 감소하면 알고리즘 수행 시간은 증가한다. 수행 시간이 증가하는 이유는 최소지지도의 감소에 따라 다량 항목에 포함되는 부분 항목집합이 많아지기 때문이다. 그러나, 본 연구에서 제안된 TARMA는 기존의 연관규칙 탐사 알고리즘 수행 결과를 입력으로 받아들이므로 연관규칙 생성 자체에 소요되는 시간은 고려하지 않고, 주어진 연관규칙이 강하게 성립하는 시간구간을 탐사하는데 소요되는 실행시간이 안정됨을 보이고자 한다. 따라서, 주어진 연관규칙에 대한 데이타베이스의 분할들에 서 임계치의 만족 여부를 확인하는 과정에서만 최소지지도가 사용되므로, 최소지지도의 변화는 TARMA의 실행 시간에 영향을 미치지 않는다.

한편, 주어진 연관규칙에 대한 긍정실례의 개수, 즉 지지도의 변화는 분할의 합병 회수를 결정하므로 전체 수행 시간에 영향을 미치는 요소이지만, T[2] 이후의 합병은 디스크에 대한 연산이 아니고, 주기억장치 상에 유지되는 전 단계의 분할테이블을 이용한다. 따라서, T[1]을 구성하기 위해 분할기준테이블을 만들면서 디스크를 스캔하는 단계에 비해 합병 회수의 증가에 따라 소요되는 비용이 상대적으로 적기 때문에 성능 분석에서 제외하였다.

입력된 연관규칙의 수를 k라고 하면 3.1절에서 제시한 TARMA를 k번 반복해서 수행하므로, 본 절에서는 단일 연관규칙에 대한 알고리즘의 성능 분석 결과만을 살펴본다. 〈표 6〉은 실험에 적용된 매개 변수들이다.

〈표 6〉 실험 매개 변수
〈Table 6〉 Experiment parameters

$\|T\|$	트랜잭션에 포함된 항목집합의 평균 길이
$\|D\|$	데이터베이스 내의 트랜잭션의 총 수
MPC	최소 분할 계수

각 트랜잭션에 포함된 항목집합의 평균 길이와 트랜잭션의 총 수는 알고리즘 수행 시간에 가장 큰 영향을 미치는 요소로서, 그 크기를 달리하여 알고리즘의 수행에 요구되는 시간의 변화가 안정됨을 보이고자 한다. 실험을 적용한 예제 데이타베이스는 각 트랜잭션에 포함된 항목집합의 평균 길이를 달리하는 두 가지 유형을 이용한다. 첫 번째 데이타베이스는

트랜잭션 당 평균 7.48개의 항목을 포함하는 트랜잭션 집합으로, 약 25000개의 트랜잭션이 1 MB의 디스크 공간을 차지한다. 두 번째는, 트랜잭션 당 평균 15개의 항목을 포함하는 트랜잭션 집합으로 약 25000개의 트랜잭션이 1.7 MB의 공간을 차지한다. 최소 분할 계수 MPC는 분할의 수와 합병 횟수를 결정하므로, 알고리즘의 수행 시간을 결정하는 중요한 요소이다. 이상에서 언급한 매개 변수를 조합하여 〈표 7〉과 같은 예제 데이타 집합을 구성하였다.

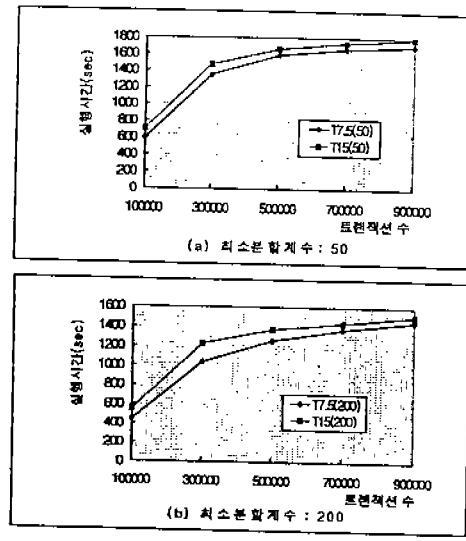
〈표 7〉 매개 변수 설정
〈Table 7〉 Parameter settings

이 름	|T|	|D|	MPC
DB1	7.48	25K~900K	10,50,100,200
DB2	15	25K~900K	10,50,100,200

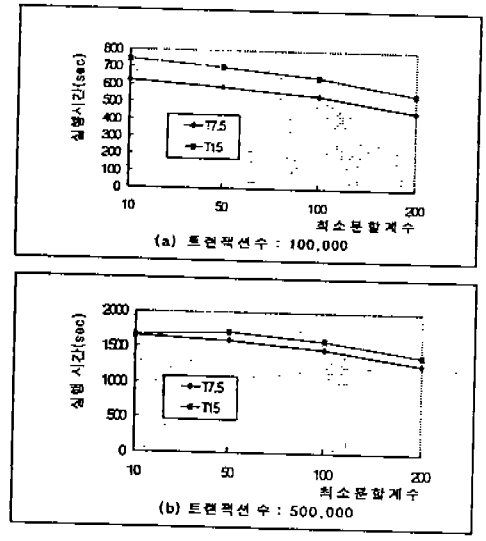
트랜잭션에 포함된 항목집합의 평균 크기가 다른 두 가지 유형(7.48과 15)의 데이타 집합에 대하여 트랜잭션의 총 수를 25,000개부터 900,000개까지 증가시키고, 각 데이타 집합에 대하여 최소 분할 계수를 10, 50, 100, 200으로 변화시켜 TARMA 수행에 소요되는 수행 시간을 정리한 결과가 〈표 8〉에 제시되었다. 〈표 8〉에서 트랜잭션 수가 25,000에서 125,000개까지 비교적 적을 경우는 수행에 소요되는 알고리즘 내의 연산(operation)의 수는 일정하지만, 기억장치 스와핑(swapping) 등의 알고리즘 실행 환경 설정에

〈표 8〉 TARMA 수행 결과
〈Table 8〉 Execution results of TARMA

트랜잭션 수	최소분할계수(50)		최소분할계수(200)	
	T7.5	T15	T7.5	T15
25000	9(sec)	12(sec)	4(sec)	5(sec)
50000	67	82	39	51
75000	238	287	167	207
100000	587	704	450	551
125000	1191	1412	962	1057
300000	1351	1474	1041	1224
500000	1589	1660	1259	1372
700000	1665	1738	1376	1435
900000	1702	1794	1455	1506



(그림 4) 트랜잭션 크기별 성능 분석
(Fig. 4) Performance analysis of transaction number



(그림 5) 최소분할계수별 성능 분석
(Fig. 5) Performance analysis of minimum partition counter

많은 영향을 받아 상당히 높은 실행 시간 증가가 나타나지만 300,000개 이상의 대용량 데이타베이스에서는 안정된 단조 증가가 나타남을 알 수 있다. (그림 6)은 트랜잭션의 수와 트랜잭션에 포함된 항목집합의

크기가 증가함에 따라 소요되는수행 시간이 단조 증가함을 볼 수 있다. (그림 7)은 최소 분할 계수가 증가되면 분할의 수가 감소되므로 실행 시간이 단조 감소함을 보여주고있다. 따라서, 제안된 TARMA 알고리즘은 현실적으로 수용 가능한 시간 비용으로 한시적 연관규칙이 성립하는 시간구간의 탐사가 가능함을 확인하였다.

5. 결 론

본 연구에서는 전체 시간 구간에 대해서는 비록 충분한 지지도와 신뢰도를 만족하지 못하지만, 부분 시간 구간에서 높은 지지도와 신뢰도를 만족하는 연관성을 한시적 연관규칙이라 정의하고, 그 타당성 척도를 제안하였으며 한시적 연관규칙이 성립되는 시간구간 값을 효율적으로 탐사하는 TARMA 알고리즘을 제시했다.

TARMA는 상대적으로 낮은 임계치를 적용한 기존의 연관규칙 탐사 알고리즘을 통해 발견된 연관규칙들에 대하여, 각 연관규칙이 강하게 성립하는 부분 시간구간을 찾는 방식으로 전개되었다. 주어진 전체 시간구간을 월, 계절 등 인위적인 크기와 무관한 임의의 부분 시간구간으로 나눌 수 있는 가짓수는 매우 많기 때문에 부분 시간구간의 탐사는 간단한 문제가 아니다. 본 논문에서는, 데이타 주도(data-driven) 방식을 채택함으로써 불필요한 시간구간을 고려하지 않도록 하였으며, 중첩된 분할을 이용하여 분할의 경계선에서 발생되는 정보의 상실을 방지하였다.

최초 분할의 형성과 각 분할의 확장에 반복적으로 이용되는 자료구조로서 분할 기준 테이블이 제시되었다. 분할 기준 테이블은 해당 연관규칙의 긍정실례를 기준으로 구성되는데 한번의 데이타베이스 스캐닝에 의해 구성되고, 원래 데이타베이스보다 그 크기가 훨씬 작아 주기억장치에 적재 가능하므로 매번 디스크 상의 데이타베이스를 스캐닝하지 않도록 하여 특히 대용량 데이타베이스에 대한 TARMA의 수행 성능을 획기적으로 향상시켰다. 따라서 현실적으로 수용 가능한 시간 비용으로 탐사가 가능하므로 응용 현장에 적용 가능한 알고리즘이다. 부분 시간구간 탐사를 위하여 먼저, 단일 연관규칙에 대한 TARMA를 제안하고, 이를 복수개의 연관규칙을 수용할 수 있도록 확장하였다.

본 논문은 최근 관심도가 증가하고 있는 연관규칙 탐사 결과의 활용 범위를 확장하여 이력(temporal) 데이타베이스 등에 관련된 의사 결정 지원에 매우 유용한 지식의 발견을 가능하게 한다. 향후 연구 방향으로는 실(real) 데이타에 의한 탐사 알고리즘 성능의 실증과 사건 발생 영역을 다차원으로 확장하여 임의의 영역에서 특별히 높은 신뢰도를 갖는 연관성의 발견 등에 관한 연구가 필요하며, 한시적 연관규칙의 탐사 결과를 사용자가 보다 이용하기 쉽게 제공할 수 있는 인터페이스의 보완이 요구된다.

참 고 문 헌

[1] W. J. Frawley, G. Piatetsky-Shapiro and C. J. Matheus, "Knowledge Discovery in Databases: An Overview", *Knowledge Discovery in Databases*, W. j. Frawley, G. Piatetsky-Shapiro Ed., AAAI Press, pp. 1-27, 1991.

[2] M. Holsheimer and A. Siebes, "Data Mining: The Search for Knowledge in Databases", Report CS-R9406, ISSN 0169-118X, CWI (Centrum voor Wiskunde en Informatica), The Netherlands, 1994.

[3] Lee Do Heon and Kim Myoung Ho, "Database Summarization Using Fuzzy ISA Hierarchies", *IEEE Trans. on Systems, Man, Cybernetics*(to be appeared), 1997.

[4] R. Agrawal, T. Imielinski and A. Swami, "Data Mining: A Performance Perspective", *IEEE Trans. on Knowledge and Data Engineering*, Vol. 5, No. 6, pp. 914-925, 1993.

[5] R. Agrawal, T. Imielinski and A. Swami, "Mining Association Rules between Sets of Items in Large Databases", *Proc. ACM SIGMOD*, Washington D.C., May, pp. 207-216, 1993.

[6] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules in Large Databases", *Proc. of 20th Intl. Conf. on VLDB*, Santiago, pp. 487-499, Sep. 1994.

[7] M. Houtsma and A. Swami, "Set-Oriented

Mining for Association Rules in Relational Databases", *Proc. ICDE*, Taipei, pp. 25-33, Mar. 1995.

[8] A. Savasere, E. Omiecinski and S. Navathe, "An Efficient Algorithm for Mining Association Rules in Large Databases", *Proc. VLDB, Zurich*, pp. 432-444, Sep. 1995.

[9] R. Srikant and R. Agrawal, "Mining Generalized Association Rules", *Proc. VLDB, Zurich*, pp. 407-419, Sep. 1995.

[10] J. Han and Y. Fu, "Discovery of Multiple-Level Association Rules from Large Databases", *Proc. VLDB, Zurich*, pp. 420-431, Sep. 1995.

[11] M. Klemettinen, H. Mannila, P. Ronkainen H. Toivonen and A. Verkamo, "Finding Interesting Rules from Large Sets of Discovered Association Rules", *Proc. CIKM*, Gaithersburg, pp. 401-407, Nov. 1994.

[12] H. Mannila and K. J. Gaiha, "Dependency Inference", in *Proc. of 3rd Intl. Conf. on VLDB*, pp. 155-158, 1987.

[13] R. Srikant and R. Agrawal, "Mining Quantitative Association Rules in Large Relational Tables", *Proc. of the ACM SIGMOD Conf. on Management of Data*, Canada, pp. 1-12, June 1996.

[14] K. H t nen, M. Klemettinen, H. Mannila, P. Ronkainen and H. Toivonen, "Knowledge Discovery from Telecommunication Network Alarm Databases", *Proc. of 12th Intl. Conf. on Data Engineering*, New Oreans, pp. 115-122, Feb. 1996.

[15] J. S. Park, M. S. Chen and P. S. Yu, "An Effective Hash-Based Algorithm for Mining Association Rules", *Proc. of the ACM SIGMOD*, pp. 175-186, May 1996.

[16] Andreas Mueller, "Fast Sequential and Parallel Algorithms for Association Rule Mining: A Comparison", CS-TR-3515, Dept. of CS, Univ. of Maryland, College Park, Aug. 1995.

[17] 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., AAAI Press/The MIT Press, pp. 307-328, 1996.

[18] P. Cohen and E. Feigenbaum, The Handbook of Artificial Intelligence, Vol. 3, William Kaufmann Pub., pp. 411-415, 1982.

[19] 이도헌, 김명호, "퍼지 개념 계층을 기반으로한 데이타베이스 속성 상호 관계의 발견", 정보과학회논문지, Vol. 22, NO. 4, pp. 514-525, April 1995.

조 일 래

1984년 전남대학교 계산통계학과(이학사)
1986년 전남대학교 대학원 계산통계학과(이학석사)
1992년 전남대학교 대학원 전산통계학과 박사과정 수료
1989년~현재 순천공업전문대학 전자계산과 조교수
관심분야: 데이타 마이닝, 데이터 웨어하우스, 이력 데이터베이스 등.

김 종 덕

1983년 전남대학교 전산학과 졸업(이학사)
1988년 국방대학원 전자계산학과(이학석사)
1995년~현재 전남대학교 대학원 전산통계학과 박사과정
관심분야: 정보통신 보안, 컴퓨터 네트워크, 객체지향 시스템 등

이 도 헌

1990년 한국과학기술원 과학기술대학 전산학과(공학사)
1992년 한국과학기술원 전산학과(공학석사)
1995년 한국과학기술원 전산학과(공학박사)
1996년~현재 전남대학교 전산학과 전임강사
관심분야: 데이타 마이닝, 데이터 웨어하우스, 퍼지 데이터베이스, 워크플로우 관리 등.