시간 데이타마이닝 프레임워크

이 준 욱[†] · 이 용 준[††] · 류 근 호[†††]

요 약

시간 데이타마이닝은 기존 데이타마이닝에 시간 개념을 추가하여 "시간값을 가진 대용량 데이타로부터 이전에 잘 알려지지는 않았지만, 묵시적이고 잠재적으로 유용한 시간 지식을 탐사하는 기술"로 정의된다. 시간 지식이란 주기적 패턴, 캘린더 패턴, 경향 등과 같이 시간 의미와 시간 관계를 가진 지식을 말한다. 실세계에서는 환자의 병력, 상품 구매 이력, 웹 로그 등과 같은 다양한 시간 데이타가 존재하며 이로부터 여러 형태의 유용한 시간 지식을 찾아낼 수 있다. 데이타마이닝에 대한 연구가 진행되면서 순차 패턴, 유사 시계열 탐사, 주기적 연관규칙 탐사 등과 같이 시간 지식을 탐사하고자 하는 시간 데이타마이닝에 대한 부분적인 연구가 수행되었다. 그러나 기존 연구는 단순히 데이타의 발생 순서 및 유사한 패턴을 찾아내는데 중점을 두고 있어 데이타가 포함하고 있는 시간 의미와 시간 관계를 탐사하는데 부족하며, 시간 지식의 전체적인 측면 보다는 연관 규칙과 같은 일부분 만을 다루고 있다는 문제점을 가지고 있다. 따라서 이 논문에서는 시간 데이타마이닝에 대한 체계적인 연구를 위하여 시간 데이타마이닝에 대한 기존 연구 내용과 해결해야 할 문제점을 분석하고 이를 바탕으로 전체적인 프레임워크를 제시하였다. 또한 그 구현 방안 및 적용평가를 수행하였다. 프레임워크에서는 시간 데이타마이닝 모델을 제안하고, 이를 바탕으로 시간 데이타마이닝 질의어와 시간 지식을 탐사할 수 있는 시간 데이타마이닝 시스템을 설계하였다.

Temporal Data Mining Framework

Jun Wook Lee[†] · Yong Joon Lee[††] · Keun Ho Ryu[†††]

ABSTRACT

Temporal data mining, the incorporation of temporal semantics to existing data mining techniques, refers to a set of techniques for discovering implicit and useful temporal knowledge from large quantities of temporal data. Temporal knowledge, expressible in the form of rules, is knowledge with temporal semantics and relationships, such as cyclic pattern, calendric pattern, trends, etc. There are many examples of temporal data, including patient histories, purchaser histories, and web log that it can discover useful temporal knowledge from. Many studies on data mining have been pursued and some of them have involved issues of temporal data mining for discovering temporal knowledge from temporal data, such as sequential pattern, similar time sequence, cyclic and temporal association rules, etc. However, all of the works treated data in database at best as data series in chronological order and did not consider temporal semantics and temporal relationships containing data. In order to solve this problem, we propose a theoretical framework for temporal data mining. This paper surveys the work to date and explores the issues involved in temporal data mining. We then define a model for temporal data mining and suggest SQL-like mining language with ability to express the task of temporal mining and show architecture of temporal mining system.

키워드 : 시간 데이터마이닝(Temporal Data Mining), 개요(Survey), 프레임워크(Framework), 시간 데이타마이닝 모델(Temporal Data Mining Model), 시간 데이타마이닝 시스템 설계(Temporal Data Mining System Design)

1. 서 론

데이타마이닝은 "데이타로부터 이전에 잘 알려지지는 않았지만, 묵시적이고 잠재적으로 유용한 지식을 추출하는 기술"로 정의되며 전자상거래, 의사결정 지원, 의료 등의 다양한 분야에서 유용하게 활용될 수 있다. 데이타마이닝에 대한 연구가 진행되면서 순차 패턴(sequential pattern), 유사 시계열 탐사(similar time sequence), 시간 연관 규칙 탐사(temporal association rule) 등과 같이 타임스탬프를 가진 시간 데이타(temporal data)로부터 시간 지식(temporal knowledge)을 탐사하고자 하는 여러 연구가 수행되었다[32, 33].

그러나 기존의 순차 패턴, 유사 시계열 탐사는 단순히 데이타의 발생 순서 및 유사한 시계열 패턴을 찾아내는데 중점을 두고 있어 데이타가 포함하고 있는 시간 의미(temporal semantics)와 시간 관계(temporal relationships)를 탐사하는데 부족하며, 시간 연관 규칙은 시간 지식의 전체적인 측면보다는 연관 규칙과 같은 일부분만을 다루고 있다는 문제점을 가지고 있다.

※ 이 연구는 한국 전자통신 연구원 우정기술 연구 센터의 연구지원과 한국과학재단 특정기초연구(R01-1999-00243) 지원으로 수행되었음.
† 준 회 원 : 충북대학교 대학원 컴퓨터 과학과
†† 정 회 원 : 한국전자통신연구원
††† 종신회원 : 충북대학교 전기전자 및 컴퓨터공학부
논문접수 : 2001년 8월 4일, 심사완료 : 2002년 3월 21일

시간 데이타마이닝(temporal data mining)은 이러한 기존 연구의 문제점을 해결하기 위해 기존 데이타마이닝에 시간 개념을 추가하여 시간 의미와 시간 관계를 가지는 유용한 시간 지식 탐사하기 위한 새로운 데이타마이닝 연구 분야이다. 시간 데이타마이닝을 통해 환자의 병력, 상품 구매 이력, 웹 로그 등과 같은 시간 데이타로부터 다양한 형태의 시간 지식을 찾아낼 수 있다. 예를 들면 구매 데이타베이스로부터 "기저귀를 사는 사람의 80%가 맥주를 산다"와 같은 기존 연관규칙 뿐만 아니라, "매년 9월~10월 동안 기저귀를 사는 사람의 80%가 맥주를 산다"와 같은 시간 의미를 가진 연관규칙을 찾아낼 수 있다. 또한 "2000년도 이전에는 A 특성을 가진 고객이 우량 고객일 가능성이 높으나 2001년 현재와 미래에는 가능성이 낮다"와 같이 시간 제약조건을 가진 분류 규칙(classification rule)을 찾아낼 수 있다.

그러나 이러한 시간 데이타마이닝의 유용성에도 불구하고 이 분야에 대한 체계적인 연구는 거의 이루어지지 않고 있다. 따라서 이 논문에서는 시간 데이타마이닝에 대한 기존 연구 내용과 해결해야 할 문제점을 분석하고 전체적인 프레임워크를 제시한다. 논문은 다음과 같은 내용으로 구성된다.

- 시간 데이타마이닝에 대한 전반적인 개요를 설명한다. 시간 데이타마이닝 개념을 정의하고 기존의 연구를 정리하여 시간 데이타마이닝 기법을 분류하였다. 또한 탐사된 지식의 관심도와 유용성을 평가하기 위한 유용성(interestingness) 평가 문제에 대해 언급하였다.
- 시간 데이타마이닝 모델을 제안한다. 시간 지식과 시간 지식의 탐사 문제를 형식화하여 정의한 후, 이를 바탕으로 시간 데이타마이닝 질의어인 TMQL(Temporal Mining Query Language)을 설계하였다. 사용자는 TMQL을 사용하여 시간 지식을 편리하게 탐사할 수 있다.
- 시간 데이타마이닝 모델을 기반으로 시간 지식을 탐사할 수 있는 시간 데이타마이닝 시스템을 설계하고 시스템을 구성하는 각 모듈의 기능을 명시한다.
- 시스템 구현을 위한 방안을 고찰하며 이 논문에서 제시한 시간 데이터마이닝 프레임워크를 기반으로하여 진행중인EC-DaMiner[47]의 시간 데이터 마이닝 시스템으로의 확장 구현 문제를 언급하고 일부 구현된 시간 관계 규칙 탐사 알고리즘의 평가 및 문제점을 제시한다. 또한 시간 지식과 관련된 적용 사례를 제시한다.

이 논문의 2장에서는 시간 데이타마이닝에 대한 개요를 설명하고, 3장에서는 시간 데이타마이닝에 대한 이론적인 모델을 제안하고, 4장에서는 시간 데이타마이닝 시스템의 구조 및 기능을 제시한다. 5장에서는 시스템 구현 방안 및 적용 평가를 제시한 후 6장에서는 결론을 맺는다.

2. 시간 데이타마이닝 개요

2.1 시간 데이타마이닝 정의

시간 데이타마이닝은 기존 데이타마이닝의 정의를 확장하여 "시간 데이타로부터 이전에 잘 알려지지는 않았지만, 묵시적이고 잠재적으로 유용한 시간 지식을 추출하는 기술"로 정의된다[33]. 만일 지식 탐사 과정에서 시간 데이타가 포함하고 있는 시간 값이 무시되거나 시간 요소(temporal component)가 고려되지 않는다면 시간 마이닝으로 간주하지 않는다.

시간 지식이란 시간 의미, 시간 관계, 시간 제약 조건을 가지는 지식이며 구체적인 예는 다음과 같다.

- 지식 1 : "매년 가을철에 임신복을 구입하는 소비자의 70%는 우산을 구입한다".
- 지식 2 : "공휴일에 냉장고를 구입하는 소비자는 대부분 카메라를 동시에 구매한다".
- 지식 3 : 해변가의 범람은 봄철 동안 높은 조류에 의해서 발생된다.
- 지식 4 : 어떤 환자들은 복합 치료약을 복용한 후 2달 후에 반응을 보이는 경향이 있다.
- 지식 5 : "비가 오기 이전 대기 압력이 강하할 확률은 60%이다".
- 지식 6 : 매주 주말에는 비디오 A를 대여한 고객의 50%가 다음에 비디오 B와 비디오 C를 대여한다.
- 지식 7 : "연초에 주가는 IBM 주식 상승 -> Intel 주식 하락 -> MS 주식 상승의 순서로 거의 매주 변동이 발생한다".
- 지식 8 : 비디오 A를 대여한 고객의 50%가 A 대여 기간 중에 비디오 B를 대여하고 B의 대여가 끝나는 즉시 비디오 C를 대여한다.

지식 1에서 알 수 있듯이 기존 연관규칙 탐사는 "임신복을 구입한 소비자가 우산을 구입한다"는 규칙은 생성할 수 있지만 "매년 가을철"이라는 시간 의미를 가진 규칙은 시간 마이닝을 통해서만 발견되어질 수 있는 규칙이다.

이와 같이 시간 지식은 연관규칙과 같은 기존 지식에 인과 관계(causal relationships) 및 시간 관계 등과 같은 시간 요소를 추가하여 확장한 지식이다.

2.1.1 시간 데이타 종류

시간 지식을 탐사하기 위해서는 시간 데이타가 필요한데 가장 적합한 것은 시간 지원 데이타베이스(temporal database)이다. 시간지원 데이타베이스는 관계형 모델과 같은 기존 데이타 모델에 시간 개념을 추가한 것으로 현재 시점의 자료 뿐만 아니라 과거에 발생된 자료를 효율적으로 관리할 수 있다[21, 38, 42-45]. 시간지원 데이타베이스는 사용자에 의해 결정되는 유효 시간(valid time) 또는 DBMS에 의해

기록되는 거래 시간(transaction time)을 저장한다.

그러나 시간 지식을 탐사하기 위해 반드시 시간지원 데이타베이스가 필요한 것은 아니며 시간값을 가지지 않은 데이타 또는 발생 시점, 시간 간격(interval) 등과 같은 시간값을 가진 기존 관계형 데이타베이스로부터 시간 지식 탐사가 가능하다.

시간 데이타의 종류는 시간값의 형태에 따라 다음 네 가지로 분류할 수 있다.

- 정적(static) 데이타 : 튜플 내에 시간값을 가지고 있지 않은 데이타로서 데이타 자체로부터 시간 지식을 탐사할 수 없고 로그 데이타에 기록된 데이타 발생 시간을 참조하여 지식을 탐사할 수 있다.
- 시퀀스 데이타 : 임의의 순서에 따라 정렬된 데이타로서 순서 번호는 가지고 있으나 타임스탬프와 같은 시간값은 가지고 있지 않은 데이타이다. 예로서 구매 순서를 키로 가진 구매 데이타베이스가 있다.
- 타임스탬프 데이타 : 시간값에 따라 정렬된 데이타로서 발생시점, 시간간격과 같은 구체적인 시간값을 가진다. 예를 들면 주식 변동 상황과 같은 시간축으로 나열된 시계열 데이타(time-series data), 환자의 병력 데이타, 위성의 기후 데이타 등이 있다.
- 완전 시간(fully temporal) 데이타 : 시간지원 데이터베이스에서 다루는 이력 데이타로서 각 튜플은 이차원 이상의 시간 차원, 즉 유효시간 또는 거래시간 값을 가진다.

2.1.2 시간 지식 분류

시간 지식의 종류는 다양하여 명확한 분류가 어려우나 시간 지식이 내포하고 있는 시간 패턴(temporal pattern)의 형태에 따라 다음과 같이 분류할 수 있다.

- 주기 패턴(cyclic pattern) : 시간이 변화하면서 주기적으로 발생하는 패턴이다. 예를 들면 앞의 예 중에서 지식1의 "매년"과 지식 7의 "매주"가 주기 패턴에 속한다.
- 캘린더 패턴(calendric pattern) : 캘린더로 표현할 수 있는 시간 패턴을 만족하는 패턴이다. 예를 들면 지식2의 "공휴일", 지식 3의 "봄철 동안", 지식 4의 "2달 후"와 같은 구체적인 시간 조건을 내포한 경우이다.
- 순차 패턴 : 특정 사건이 순차적으로 연이어 발생하는 패턴으로 한 사건이 다른 사건 발생의 원인이 되는 인과 관계를 가진다. 지식 5와 지식 6이 이에 속한다.
- 경향(trend) : 시계열 데이타에서 시간이 흐르면서 값이 변화하는 패턴으로 향후 변동을 예측할 수 있다. 예를 들면 지식7에서 "IBM 주식의 상승 -> Intel 주식 하락 -> MS 주식 상승"과 같은 경우이다.
- 시간 관계 : 시간간격을 가진 여러 사건 간의 관계를 표현한 지식이다. 예를 들면 지식8에서 "비디오 A를 대여하는 기간 중에 B를 대여하는 경우이다". 지식 6의 순차 패턴과 달리 단순히 사건이 종료된 후에 다음 사건이 발생하는 것이 아니라, 여러 사건이 특정 기간 동안 중복되어 발생한다.

이러한 시간 지식은 한 패턴만을 내포하지 않고 여러 종류의 패턴을 복합적으로 가질 수 있다. 예를 들면 지식 6은 주기 패턴과, 순차 패턴을 모두 포함한 지식이고 지식 7은 주기패턴과 경향을 모두 포함하고 있다.

2.2 시간 데이타마이닝 기법

시간 데이타로부터 시간 지식을 탐사하기 위한 시간 마이닝 기법에 대한 여러 연구가 진행되었다[7, 13, 28, 34, 41]. 이러한 기존 연구들은 크게 시간 규칙(temporal rules) 탐사, 시퀀스 마이닝(sequence mining), 경향 분석(trend analysis) 등으로 분류된다.

시간 규칙 탐사는 연관규칙 탐사, 분류 등의 기존 데이타 마이닝 기법을 주기 패턴, 캘린더 패턴 등과 같은 시간 패턴을 탐사하도록 확장한 기법이다. 시퀀스 마이닝은 시퀀스 데이타로부터 순차 패턴과 같은 인과관계를 탐사하는 기법이다. 경향 분석은 시계열 데이타로부터 여러 형태의 경향을 탐사하는 기법이다.

2.2.1 시간 규칙 탐사

- 시간 연관규칙 탐사(temporal association)

기존 연관규칙을 시간 패턴을 탐사하도록 확장한 기법이다. 여기에 속한 기법으로는 주기적으로 반복되는 연관규칙을 탐사하는 주기적 연관 규칙 탐사(cyclic association)[27], 캘린더로 표현된 시간 패턴을 가지는 연관규칙을 탐사하는 캘린더 연관규칙 탐사(calendric association)[14, 30]등이 있다.

이 기법에서 해결해야 할 문제는 모든 시간 단위(granularity)에 대해 모든 연관규칙을 탐사할 필요가 없는 보다 효율적인 알고리즘을 설계하는 문제이다.

- 시간 분류(temporal classification)

분류는 데이타를 각 클래스가 갖는 특징에 근거하여 분류하는 기법으로 트레이닝 데이타로부터 분류 규칙을 생성한다[2]. 분류 규칙을 표현하기 위한 방법으로는 의사 결정 트리(decision tree)가 가장 많이 사용된다. 시간 분류는 기존 분류 기법에 규칙의 유효 기간과 같은 시간 제약 조건을 추가하여 확장할 수 있다.

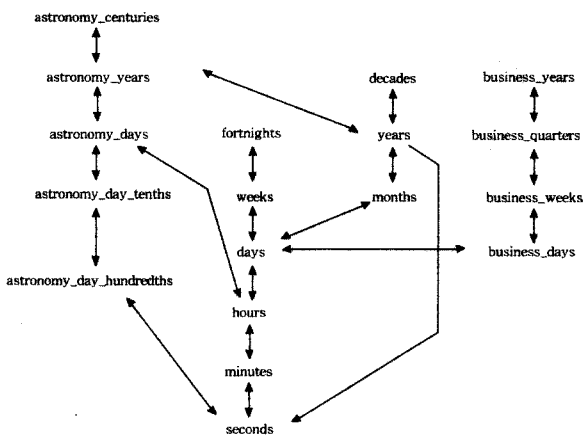
예를 들어 신용카드 회사에서 우량고객과 불량고객 판단 시 고객의 성별, 소득, 나이 등의 평가 항목을 추출하여 분류 규칙을 생성하는 경우 "2000년 이전의 가입 고객"과 같은 캘린더로 표현할 수 있는 시간 제약 조건을 부여하여 분류를 수행할 수 있다. 이러한 시간 분류규칙은 2000년 이전과 같은 과거 시점의 규칙이 현재나 미래에는 틀릴 수 있다는 점에서 유용하다. 이 기법에서 해결해야 할 문제는 고객의 나이, 생년월일 등과 같은 시간값으로부터 각 클래

스의 특징을 정확히 분류할 수 있는 시간 분류 알고리즘을 개발하는 것이다.

● 시간 특성화(temporal characterization)

특성화는 데이타 집합의 일반적 특성을 분석하는 기법으로 일반화(generalization) 과정에 의한 자료 요약 과정(summarization)을 거쳐 특성 규칙(characterization rule)을 탐사한다[2]. 규칙 탐사 과정에서 각 튜플의 항목값을 ISA 계층을 이용하여 상위 개념의 값으로 대치하여 요약한다. 시간 특성화는 기존 특성화 기법에 시간 일반화 개념을 추가하여 확장할 수 있다. 즉, 시간값을 가진 항목을 (그림 1)과 같은 시간 계층(time hierarchy)을 이용하여 요약한다.

이 기법에서 해결해야 할 문제는 응용 분야별로 다른 시간 계층을 고려하는 문제이다. 예를 들어 회계 업무의 시간 계층이 "일 -> 주일 -> 분기 -> 회계 년도"인 경우와 일반적인 시간계층이 "일 -> 월 -> 년도"인 경우는 서로 다른 특성화 결과가 나올 수 있다.



(그림 1) 시간 계층

그 외에도 클러스터링과 같은 다른 데이타마이닝 기법도 시간요소를 추가하여 시간 규칙 탐사 기법으로 확장할 수 있다.

2.2.2 시퀀스 마이닝

시퀀스 마이닝의 대표적인 연구는 순차패턴 탐사이다. 순차패턴 탐사는 항목집합으로 구성된 트랜잭션들 간에 특정 항목집합이 순차적으로 발생하는 패턴을 탐사하는 기법으로 한 트랜잭션 내에서 발생하는 항목들 간에 연관성을 탐사하는 연관규칙에 시간적인 관계를 추가한 것이다. 즉, 고객 트랜잭션 내에 항목 A가 존재한다면, 그 이후에 발생하는 같은 고객의 다른 트랜잭션 내에 항목 B, C, … 가 차례로 존재한다는 시퀀스를 탐사하는 기법이다. 각 고객들의 트랜잭션을 시간순서로 볼 수 있는데 이를 고객 시퀀스라고 부른다. 순차패턴 문제는 사용자가 지정한 최소 지지도(minimum support threshold)를 만족하는 모든 시퀀스들 사이에서 최대 시퀀스를 탐사하는 것이다. 대표적인 순차패턴 알고리즘 AprioriAll, AprioriSome은 연관규칙 알고리즘인 Apriori를 기반으로 하고 있다[3, 5].

빈발 에피소드(frequent episode) 탐사[24, 25]는 일련의 사건 시퀀스(event sequence) 데이타로부터 빈번하게 발생하는 에피소드를 찾는 순차패턴 기법이다. 에피소드는 빈번하게 발생하는 특정 사건 시퀀스로 정의되며 시퀀스를 구성하는 사건은 서로 밀접하게 관련된 사건이다. 탐사 문제는 사용자가 지정한 윈도우 크기를 갖는 시간 윈도우들의 집합에서 에피소드가 발생한 윈도우의 비율이 최소 발생도(frequency threshold) 이상을 만족하는 모든 빈발 에피소드를 찾는 것이다. 예를 들어 매초 마다 발생한 사건 시퀀스를 (A, B, D, A, C, B, D, A, D), 시간 윈도우 크기를 3초, 최소 발생도를 50%라 할 때 (A, B)는 세 개의 윈도우 중에 두개의 윈도우 내에서 발생하였으므로 에피소드이다. SPIRIT[18]는 사전에 정의된 제약조건(regular expression constraint) R을 만족하는 모든 빈발 시퀀스를 찾는 기법이다.

이러한 순차패턴 기법에 시간 요소를 추가하여 확장한 기법으로는 GSP(Generalized Sequential Pattern)[39], 주기적 순차패턴 탐사[20], 예기치 않은 패턴(unexpected pattern) 탐사[9] 등이 있다. GSP는 AprioriAll 같은 기존 알고리즘에 시간 제약 조건, 슬라이딩 시간 윈도우, 분류를 이용한 일반화 개념을 추가한 기법이다. 주기적 순차패턴 탐사는 주기적으로 발생하는 순차패턴을 탐사하는 기법이다. 예기치 않은 패턴탐사는 시간 논리(temporal logic) 표현식에 의해 사전에 정의된 시간패턴 P를 찾는 기법이다. 이 기법에서는 시간패턴 P의 발생 횟수가 사용자가 지정한 기대임계치(thresholds)를 초과한다면 P는 예기치 않은 유용한 패턴이라고 간주한다.

이러한 기존 연구들은 주로 순차패턴 탐사에 대해 수행되었으나 현실세계에는 순차패턴 외에도 시퀀스 데이타로부터 찾을 수 있는 보다 다양한 형태의 시간 패턴 및 인과관계가 존재하므로 이에 대한 연구가 필요하다.

2.2.3 경향 분석

주식, 물가, 판매량 등과 같이 데이타가 시간축으로 나열된 시계열 데이타로부터 경향을 파악하여 미래를 예측하는 기법이다. 대표적인 경향분석 기법으로는 유사 시계열 탐사와 예외 분석(deviation analysis)이 있다. 유사 시계열 탐사는 여러 다른 패턴들 중에서 서로 유사한 패턴을 찾는 기법이다[4, 6]. 예를 들어 주식 동향을 나타내는 시계열 데이타로부터 특정 주식과 비슷한 형태의 데이타를 찾아낼 수 있다면 두 주식의 운영 전략에 유사성을 부여할 수 있게 된다. 탐사 문제는 주어진 시계열 시퀀스 집합으로부터 사용자가 질의한 질의 시퀀스(query sequence)와 유사한 모든 시퀀스 또는 유사 시퀀스(similar sequence)의 모든 쌍을 탐사하는 것이다. 유사 시퀀스의 쌍을 탐사하는 기법은 전반적으로 유

사한 시퀀스 쌍을 탐사하는 기법과 부분적으로 유사한 시퀀스 쌍을 탐사하는 기법[17]으로 분류된다. 또한 공간 데이타, 이미지 데이타 등을 위한 유사 시계열 탐사 기법이 연구되었는데 공간 데이타에서 빠른 유사 조인(similarity join)을 위한 알고리즘 및 색인 구조가 제안되었다[35].

예외 분석(deviation analysis)은 outliers라고도 하며 유사 시계열 탐사와 달리 일반적인 경향을 따르지 않는 예외적인 패턴을 찾아내는 기법이다[31]. 예를 들면 일정 기간 동안 대부분의 주식들에서 볼 수 있는 일반적인 주가 등락 경향과 다른 경향을 보이는 예외적인 주식을 탐사하여 원인을 분석할 수 있다.

이러한 경향 분석은 이미지 처리, 패턴 분석 등과 같은 타 분야에서도 많이 다루고 있는 연구 분야로서 독립된 연구 분야를 차지하고 있다.

2.2.4 기타 기법

시간 데이타마닝 기법은 앞에서 분류한 기법 외에도 여러 다양한 기법이 존재한다. 예를 들면 규칙 마이닝(rule mining)이란 기존에 탐사된 규칙들의 집합으로부터 유용한 메타 규칙(meta rule)을 찾아내는 기법이다[33]. 즉, 동일한 연관규칙이라도 지지도나 발생도는 시간에 따라 변화하는 이력(history)을 가지므로 이력으로부터 변화 패턴을 파악함으로써 보다 고수준의 규칙 탐사가 가능하다.

앞에서 언급한 바와 같이 시간 지식을 탐사하기 위해 반드시 시간지원 데이타베이스가 필요한 것은 아니며 기존 데이타베이스로부터도 탐사가 가능하다. 그러나 모든 데이타로부터 모든 시간 지식을 탐사할 수 있는 것이 아니라 시간 데이타의 종류에 따라 탐사할 수 있는 시간 지식과 사용할 수 있는 시간 마이닝 기법이 제한을 받는다. <표 1>은 시간 데이타와 시간 마이닝 기법간의 이러한 관계를 요약하여 정리한 것이다.

〈표 1〉 시간 데이타와 사용 가능한 시간 마이닝 기법

마이닝기법 \ 시간 데이타	정적 데이타	시퀀스 데이타	타임스탬프 데이타	완전 시간 데이타	기존에 탐사된 규칙
기존 기법	가 능	불가능	가 능	가 능	불가능
시간연관규칙탐사	불가능	일부 가능 (연관규칙 내의 항목 간의 발생 순서)	가 능	가 능	불가능
시간 분류	불가능	가 능	가 능	가 능	불가능
시간 특성화	불가능	가 능	가 능	가 능	불가능
시퀀스 마이닝	불가능	가 능	가 능	가 능	불가능
경향 분석	불가능	가 능	가 능	가 능	불가능
규칙 마이닝	불가능	불가능	불가능	불가능	가 능

2.3 지식의 유용성 평가

기존 데이타마이닝 기법들은 너무 많은 필요 이상의 규칙을 생성하거나 사용자의 의도와 관계없는 정보이어서 그 유용성이 떨어질 수 있다. 따라서 연관규칙 탐사에서는 지지도와 신뢰도의 개념을 도입하여 지식의 유용성을 평가하였다. 지지도는 연관규칙 x->y에서 전체 트랜잭션 개수 N에 대한 x와 y를 만족하는 트랜잭션 개수의 비율이며, 신뢰도는 x를 만족하는 트랜잭션에 대한 y를 만족하는 트랜잭션의 비율이다. 지지도와 신뢰도는 트랜잭션의 발생 빈도가 많을수록 유용하다는 전제하에 만들어진 측정 지표이다.

그러나 현실 세계에서는 발생 빈도와 상관없이 유용한 지식이 존재하는 경우가 많으며 다양한 측정 지표가 만들어 질 수 있다. [37]에서는 통계적으로 유용한 지식이 항상 유용한 것은 아니라고 하였으며 사용자의 주관적 측정 지표를 제안하였다.

시간 데이타마이닝 분야에서도 앞에서 언급한 효율적인 시간 마이닝 기법을 연구하는 문제 외에도 시간 지식의 유용성 평가에 대한 연구가 매우 중요하며 특히 시간 데이타의 특성에 적합한 측정 지표가 필요하다. 시간 데이타마이닝에서의 유용성은 크게 규칙 자체에 대한 시간 유용성과 시간에 따라 지속적으로 갱신되는 규칙의 관리 문제로 분류할 수 있다.

2.3.1 시간 유용성

탐사된 지식의 유용성 평가에 있어서 시간 특성을 고려한 여러 연구가 수행되었다.

GSP 에서는 항목간의 인접성(proximity) 평가를 위한 시간 제약 조건과 시간 윈도우의 개념이 제안되었다[39]. 기존 순차패턴에서는 항목 A가 발생한 후 항목 B가 발생하는 시간 간격의 크기를 고려하지 않으므로 유용성이 상실되는 경우가 발생한다. GSP는 사용자가 미리 지정한 시간 제약 조건(시간간격 범위)을 만족하는 순차패턴 만을 탐색함으로써 이러한 문제를 해결하였다. 또한 순차패턴을 구성하는 임의의 한 항목 집합(Itemset) 이 n개의 다른 트랜잭션 내에 존재하는 항목들로 구성되었다고 하더라도 n개의 트랜잭션이 시간 윈도우내에서 발생하였다면 같은 항목집합에 속한 것으로 간주한다.

또한 장 바구니 분석(market basket analysis)을 위해 시계열 데이타로부터 의외의 놀라운 패턴(surprising pattern)을 탐색하는 연구가 수행되었다[11]. 기존의 연관규칙 탐사 시간에 따라 변화하는 항목의 발생 빈도를 고려하지 않는데 비해, 이 연구는 시계열 데이타를 사용자가 지정한 윈도우 크기에 따라 세그먼트들로 분리한 후 세그먼트 마다 다른 항목의 발생 분포를 파악하여 전체적으로는 발생 빈도가 낮지만 특정 윈도우에서 발생 빈도가 높은 놀라운 연관규칙을 탐사할 수 있다.

예기치 않은 패턴 탐사에서는 시간패턴 P의 발생 횟수가 사용자가 지정한 기대 임계치를 초과한다면 P는 예기치 못한 패턴으로서 유용하다고 간주하고 이를 찾아내는 문제는 NP-complete하다는 것을 증명하였다[9].

2.3.2 규칙 관리

규칙 자체의 시간 유용성 측면 외에도 탐사된 규칙의 유효 기간에 대한 평가가 필요하다. 즉, 과거에 탐사된 규칙은 과거시점에는 옳으나 현재, 미래에는 틀릴 수 있으므로 규칙은 지속적으로 갱신, 관리되어야 한다. 이러한 규칙 관리 문제는 특히 점진적 마이닝(incremental mining)에 대한 연구에서 집중적으로 다루어졌다.

[15]는 연관규칙 탐사에서 추가된 트랜잭션과 삭제된 트랜잭션에 대해서만 항목의 지지도와 신뢰도를 계산한 뒤, 변경된 최종 데이타베이스의 빈발 항목에 적용함으로써 연관규칙을 갱신하는 점진적 마이닝 기법을 제안하였다. 이 방법은 데이타베이스 갱신이 발생할 때마다 전체 데이타베이스에 대해 연관규칙 탐사를 반복해야만 하는 비용을 줄이기 위해 제안되었다. 또한 데이타웨어하우스와 같이 데이타 삽입 트랜잭션 만이 주로 발생하는 경우에 적용할 수 있는 알고리즘이 제안되었다[16].

[29]는 시간 정보를 이용하여 관련된 데이타에 대해서만 마이닝 작업을 수행할 수 있는 점진적 연관규칙 탐사 알고리즘을 제안하였다. 전체 데이타베이스에 대해 연관규칙 탐사를 수행하는 것이 아니라 사용자가 미리 지정한 시간 윈도우 내에서 새로 발생한 트랜잭션들에 대해서만 탐사를 수행 한 후, 기존의 규칙과 새로 탐사한 규칙을 비교하여 기존 규칙이 시간 윈도우 내에서도 탐사되었다면 강성 규칙(strong rules)으로 판단하여 보전하고, 아니면 삭제하는 알고리즘이다.

2.4 향후 연구 주제

2.4.1 시간 데이타 마이닝 시스템

시간 데이타마이닝의 중요성에도 불구하고 현 시점에서 아직 뚜렷한 시스템은 개발되어 있지 않으나 특정 분야에 활용하기 위한 시간 데이타마이닝 기법은 지속적으로 연구되었다.

예를 들어 의료 분야에서 환자의 투약 이력 데이타로부터 시간 지식을 탐사하는 연구가 진행되었으며 이를 통해 특정 종류의 약품들은 일정 기간 동안 동시에 투약할 경우 부작용을 낳을 수 있다는 유용한 지식을 탐사할 수 있다[40]. RX 프로젝트[10]는 환자의 병력과 같은 시퀀스 데이타로부터 증상간의 복잡한 인과 관계를 탐사하는 마이닝 기법을 개발하였다. 그 외에도 다양한 응용 분야에서 시간 지식을 탐사하기 위한 연구가 진행되었다.

그러나 이러한 연구들은 상기에서 분류한 시간 마이닝 기법의 일부분 만을 구현하고 있을 뿐만 아니라 데이타 화일을 대상으로 개발된 기법이어서 대용량 시간 데이타에 적용하기에는 많은 문제점을 가지고 있다.

따라서 주요시간 마이닝 기법을 모두 지원하고 대용량 데이타로부터 시간 지식을 효율적으로 탐사할 수 있는 시간 데이타마이닝 시스템에 대한 연구개발이 필요하다. 이를 위해서는 첫째, 기존 DBMS와의 통합이 연구되어야 하며 특히 효율적인 데이타마이닝 작업을 위해 DBMS에서 표준 데이타마이닝 연산자들이 지원될 필요가 있다[36]. 둘째, 대용량 시간 데이타를 효율적으로 저장하고 검색할 수 있는 저장 구조, 특히 시간 색인(temporal indexing) 기법[38]과 이를 기반으로 한 시간 마이닝 알고리즘에 대한 연구가 필요하다. 마지막으로 다양한 의미와 구조를 갖는 시간지식의 가시화에 대한 연구가 필요하다.

2.4.2 시간관계 규칙 탐사

기존 시간 데이타마이닝 연구는 트랜잭션의 발생 시점(time point)만을 가진 시간 데이타를 다루고 있으며 시간 간격(time interval)을 가진 데이타는 거의 고려하고 있지 않다. 그러나 실세계에서는 환자의 병력, 상품 구매 이력, 웹 로그 등과 같은 다양한 시간간격 데이타가 존재하며 이로부터 여러 유용한 지식을 찾아낼 수 있다. 예를 들면 시간간격을 가진 구매 트랜잭션으로 구성된 데이타베이스로부터 "비디오 A를 대여한 고객의 50%가 다음에 비디오 B와 비디오 C를 대여한다"와 같은 순차 패턴뿐만 아니라, "비디오 A를 대여한 고객의 50%가 A 대여 기간 중에 비디오 B를 대여하고 B의 대여가 끝나는 즉시 비디오 C를 대여한다" 같은 유용한 시간 관계를 찾아낼 수 있다. Allen은 시간간격 사이에 발생할 수 있는 시간관계와 시간관계를 구할 수 있는 시간간격 연산자(interval operator)를 정의하였다[8, 23, 26].

따라서 시간간격 연산자를 기반으로 시간간격 데이타로부터 시간관계 규칙(temporal relation rule)을 효율적으로 탐사할 수 있는 기법에 대한 연구가 필요하다.

2.4.3 다중 시간 모델

시간 의미는 각기 다르게 해석될 수 있으므로 분야 마다 다른 여러 시간 모델들이 존재한다. 예를 들면 시간은 연속성(continuous), 불연속성(discrete), 선형(linear) 등과 같은 다른 성질을 가질 수 있으며 그레고리안 달력, 이슬람 달력 등과 같은 다른 형태의 달력들이 존재한다. 그러나 대부분의 기존 시스템들은 특정 분야에 대한 시간 모델만을 다룬다. 따라서 특정 모델에 종속되지 않고 여러 다른 시간 모델을 처리할 수 있는 범용성있는 시간 마이닝 기법에 대한 연구가 필요하다.

이러한 연구 주제 외에도 시간 지식의 유용성 평가 문제, 시간단위(granularity) 해석 등에 대한 보다 심도 있는 연구가 필요하다.

3. 시간 데이타마이닝 모델

시간 데이타마이닝은 현시점에서 체계적인 연구가 거의 이루어지지 않고 있으므로, 이 장에서는 시간 데이타마이닝 문제를 명확히 정의하기 위한 시간 데이타마이닝 모델을 제안한다. 그리고 앞에서 언급한 시간 마이닝 기법 중에서 경향 분석은 독립된 연구 분야이므로 제외하며 시간 규칙

탐사와 시퀀스 마이닝 만을 대상으로 한다.

먼저 시간지식 탐사 문제를 정의한 후, 이를 바탕으로 시간 데이타마이닝 질의어인 TMQL을 설계하고 시간 지식 탐사를 위한 모델을 제안한다.

3.1 문제 정의

3.1.1 시간 표현식

시간 지식은 시간 의미와 시간 제약 조건을 가지는 지식이다. 따라서 시간 지식을 명확히 표현하기 위해서는 시간 패턴을 표현할 수 있는 표현식이 필요하다.

시간표현식 TimeExp는 [12]에서 제안한 캘린더 대수(calendar algebra)를 주기적 패턴을 포함하도록 확장하여 표현한다. 여기서 캘린더는 시간간격의 집합으로 정의된다. TimeExp는 시간간격을 형식화하여 표현한 캘린더 표현식(calendar interval expression)과 주기적 패턴을 표현한 주기 표현식(periodic expression)으로 구성된다.

정의 3.1 : 캘린더 표현식 $CI := \prod_{i=1}^{m} U_i(r_i), m \geq 1, U_{i+1} < U_i$이다. 여기서 U_i는 시간단위, r_i는 U_i에 대한 순서 번호, U_m은 최소 시간단위이다. r_i가 명시되지 않으면 U_i 도메인에 속한 모든 시간을 의미한다.

CI는 단일 시점(time point)과 시간간격의 집합을 모두 표현할 수 있다. 예를 들어 캘린더 "Years(2001).Months(1).Days(26)"은 "2001년 1월 26일"을 의미하고, "Years(2001).Months(1).Days"은 시간간격 "2001년 1월 1일~1월 31일"을 의미한다. 또한 "Years(2001).Months(1).Weeks.Days"는 "2001년 1월의 주별 기간"을 의미하며 최소 시간단위는 Days이므로 시간간격 집합 {(-1, 6), (7, 13), (14, 20), (21, 27), (28, +3)}으로 변환할 수 있다. -1은 2000년 12월 31일, +3은 2001년 2월 3일을 뜻한다.

임의의 시간간격을 가진 두 캘린더를 $CI_1 = (vs_1, ve_1)$, $CI_2 = (vs_2, ve_2)$라고 하면 CI_1과 CI_2사이에는 시간 관계를 가질 수 있다[8]. 여기서 vs는 시간간격의 시작점, ve는 종료점이다.

정의 3.2 : 캘린더 시간관계 $R(CI_1, CI_2) = \{P(CI_1, CI_2) \mid P \in IO\}$, $CI_1 \neq CI_1$이다.

시간간격 연산자의 집합은 $IO = \{before, equal, meets, overlaps, during\}$이고 $P(CI_1, CI_2)$는 CI_1, CI_2간의 시간 관계를 표현하는 이진 술어(predicate)이다. $R(CI_1, CI_2)$는 다음과 같이 수학적으로 표현할 수 있다.

$$before(CI_1, CI_2) \equiv ve_1 < vs_2$$

$$equal(CI_1, CI_2) \equiv (vs_1 = vs_2) \wedge (ve_1 = ve_2)$$

$$meets(CI_1, CI_2) \equiv ve_1 = vs_2$$

$$overlap(CI_1, CI_2) \equiv (vs_1 < vs_2) \wedge (ve_1 > ve_2)$$

$$during(CI_1, CI_2) \equiv (vs_1 > vs_2) \wedge (ve_1 < ve_2)$$

예를 들어 시간관계 "***overlap***(Years(2001).Months(1).Weeks. Days,Years(2001).Months(1))"는 "2001년 1월의 주별 기간 중에서 1월에 포함되는 기간"을 의미하며, 시간간격 집합 {(1, 6), (7, 13), (14, 20), (21, 27), (28, 31)}로 변환할 수 있다.

또한 임의의 시점을 가진 두 캘린더 CI_1과 CI_2로 연속된 기간(contiguous time interval)을 표현할 수 있다. 모든 캘린더 표현식의 집합을 CIS이라고 한다.

정의 3.3 : 연속 기간은 $[CI_1^+, CI_2^+]$이다. 단, $CI_1^+ \in CIS \cup \{-\infty\}$, $CI_2^+ \in CIS \cup \{+\infty\}$이다.

예를 들어 [Years(2001).Months(1).Days(1), Years(2001).Months(1).Days(31)]은 "2001년1월1일~1월31일" 까지의 기간을 표현한다.[-∞, Years(2001).Months(1).Days(1)]은 "2001년 1월 1일 이전", [Years(2001).Months(1).Days(1), +∞]는 "2001년 1월 1일 이후"를 표현한다.

정의 3.4 : 주기 표현식 $PT := U_0 \cdot \prod_{i=1}^{m-1} U_i(r_i) \cdot U_m(a_m : b_m)$, $m \geq 1$, $U_{i+1} < U_i$이다. 여기서 r_i, a_m, b_m은 U_i에 대한 참조 번호이고, U_0은 주기 단위(cycle unit)이다.

예를 들어 "Weeks.Days(2).Hours(12 : 13)"의 주기 표현식은 "매주 월요일(2번째 요일) 12시~13시"를 표현하며 주기단위는 Weeks이다.

상기의 정의들로부터 시간표현식 TimeExp을 정의한다. 모든 캘린더 표현식의 집합을 CIS, 캘린더 시간관계의 집합을 CRS, 연속 시간구간의 집합을 TIS, 주기표현식의 집합을 PTS라고 하면 TimeExp는 다음과 같이 정의된다.

정의 3.5 : 시간표현식 $TimeExp := PT^+$ for CI^+이다. 여기서 φ $PT^+ \in PTS \cup \{\varphi\}$, $CI^+ \in CIS \cup CRS \cup TIS$이다.

즉, 시간표현식은 캘린더표현식, 캘린더 시간관계, 기간으로 구성된 시간 패턴 동안에 발생하는 시간 주기를 간략하게 표현할 수 있으며 각 예는 다음과 같다.

- Days.Hours[12 : 13] for Years(2001).Months.Weeks[1] : "20001년 첫 번째 주 동안 매일 12시~13시"
- Days.Hours[12 : 13] for overlap(Years(2001).Months(1).Weeks.Days,Years(2001).Months(1)) : "2001년 1월의 주별 기간 중에서 1월에 포함되는 기간 동안 매일 12시~13시"
- Weeks.Days(1 : 3) for [Years(2001).Months(1),Years(2001).Months(12)] : "2001년 1월~12월 동안 매주 월요일~수요일"
- for [Years(2001).Months(1),Years(2001).Months(12)] : "2001년 1월~12월 동안"

3.1.2 시간 지식 탐사

시간 지식은 시간 의미와 시간 제약 조건을 가지는 지식으로 정의한다. 시간 의미란 시간 지식이 내포하고 있는 시간 패턴을 의미하며, 시간제약 조건이란 지식이 만족해야 할 지식의 유용성을 의미한다. 시간 지식은 기존 데이타마이닝 규칙을 확장하여 다음과 같이 정의된다.

정의 3.6 : 시간 지식은 <Rule, TimeExp> 형태로 표현된다. Rule은 연관규칙, 분류규칙 등과 같은 기존 데이타마이닝 규칙 형식과 동일하며, TimeExp는 시간 의미 및 시간 제약조건을 명시한다. 임의의 Rule은 TimeExp를 만족해야 한다.

시간 지식 탐사는 임의의 주어진 시간 데이타로부터 시간지식을 탐사하는 문제로서 다음과 같이 정의한다.

정의 3.7 : 시간지식 탐사는 입력 데이타베이스 로부터, 미리 지정한 임계치를 만족하는 모든 시간지식 <Rule, TimeExp>을 탐사하는 문제이다.

임계치는 지지도, 신뢰도와 같은 기존 임계치 외에도 앞의 시간 지식 유효성 문제에서 인접성, 시간 윈도우 크기 등을 모두 포함한다.

3.1.3 시간 지식 탐사 예

정의 3.6과 정의 3.7을 보다 구체화하여 시간 데이타마이닝 기법 별로 시간 지식 탐사 문제를 정의하면 다음과 같다.

- 시간 연관규칙 탐사

$I=\{i_1, i_2, \cdots, i_n\}$를 모든 항목의 집합이라 하고 트랜잭션들로 구성된 데이타베이스를 D라고 한다. 트랜잭션 S는 트랜잭션-ID, 항목집합, 트랜잭션이 발생한 타임스탬프로 구성되며 $S=\langle tid,\ itemset,\ tiemstamp\rangle$, $itemset \in I$로 표현한다.

정의 3.8 : 시간 연관규칙은 <AssoRule, TimeExp> 형태로 표현된다. AssoRule은 $X \Rightarrow Y$ 형식을 가지며 $X \subset I$, $Y \subset I$, $X \cap Y = \phi$이다. TimeExp는 정의 3.5의 시간표현식으로 시간간격의 집합이며 $\Phi(TimeExp)=\{p_1, p_2, \cdots p_n\}$으로 해석된다. 여기서 p_i는 시간간격이다.

예를 들면 대표적인 시간 연관규칙인 주기적 연관규칙과 캘린더 연관규칙은 기존 연관규칙을 확장하여 다음과 같이 표현된다. <등산화 => 내복, Years.Months(3 : 5) ***for*** [Years(1998),Years(2000)]>은 주기적 연관규칙으로 "1998년~2000년 동안 매년 3월~5월에 등산화를 구매하는 사람은 내복을 구매한다"를 의미하며, Years.Months(3 : 5) ***for*** [Years(1998),Years(2000)]은 1998년 1월을 시작점으로 하는 시간간격의 집합 {(3, 5), (15, 17), (27, 29)}로 해석된다.

또한 <등산화 => 내복, ***for overlap***(Years(2001).Months(1).Weeks.Days,Years(2001).Months(1))>은 캘린더 연관규칙으로 "2001년 1월의 주별 기간중에서 1월에 포함되는 기간 동안 등산화를 구매하는 사람은 내복을 구매한다"를 의미한다.

시간 연관규칙에서는 기존의 지지도, 신뢰도 외에도 유용성 문제에서 언급한 발생도라는 새로운 임계치를 사용한다. 임의의 시간간격 p에 존재하는 데이타베이스 D의 부분집합을 $D(p)$라고 한다. 단, $p \in \Phi(TimeExp)$이다.

앞서 서론에서 언급되었던 8개의 시간 지식유형을 시간 연관규칙으로 분류하면 <표 2>와 같다.

<표 2> 시간연관규칙(temporal association)의 분류

주기패턴 (cyclic pattern)	시간이 변화하면서 주기적으로 발생하는 패턴이다.	예. 지식 1 지식 7
캘린더 패턴 (calendric pattern)	캘린더로 표현할 수 있는 시간 패턴을 만족하는 패턴이다.	예. 지식 2 지식 3 지식 4
순차패턴	특정 사건이 순차적으로 연이어 발생하는 패턴으로 한 사건이 다른 사건 발생의 원인이 되는 인과 관계를 가진다.	예. 지식 5 지식 6
경향(trend)	시계열 데이타에서 시간이 흐르면서 값이 변화하는 패턴으로 향후 변동을 예측할 수 있다.	예. 지식 7
시간관계	시간간격을 가진 여러 사건 간의 관계를 표현한 지식이다.	예. 지식 8

정의 3.9.1 : $D(p)$내에 최소 지지도 S_{min}과 최소 신뢰도 C_{min} 이상을 만족하는 임의의 AssoRule이 존재한다면, <AssoRule, TimeExp >는 p를 만족한다.

정의 3.9.2 : <AssoRule, TimeExp>가 $\Phi(TimeExp)$에 속한 시간간격의 f% 이상을 만족한다면, AssoRule은 TimeExp를 만족한다. 이때 f%를 최소 발생도(minimum frequency) F_{min}이라고 정의한다

예를 들어 연관규칙 <등산화 => 내복>이 시간표현식 "Years.Months(3 : 5) ***for*** [Years(1998),Years(2000)]>"가 포함하는 {(3, 5), (15, 17), (27, 29)} 중에서 (3, 5), (15, 17) 동안에 최소지지도와 최소신뢰도 이상을 만족한다면 발생도는 66.6%이고, F_{min}이 50%라면 이 규칙은 시간 표현식을 만족한다.

정의 3.10 : 시간 연관규칙 탐사는 데이타베이스 D로부터, 사용자가 미리 지정한 최소 발생도 F_{min}에 대하여 최소 지지도 S_{min}과 최소 신뢰도 C_{min}을 만족하는 규칙 $\langle X \Rightarrow Y,\ \langle X \Rightarrow Y, TimeExp\rangle$를 탐사하는 문제이다.

결론적으로 시간 연관규칙은 발생도를 고려함으로써 시간에 따라 변화하는 연관규칙을 탐사할 수 있다.

- 시간 분류 규칙 탐사

분류를 위해 데이타베이스 D로부터 추출한 트레이닝 데이타 집합을 TS라고 한다. TS를 구성하는 트랜잭션 S는 트랜잭션-ID, 항목집합, 클래스 라벨, 트랜잭션이 발생한 타임

스탬프로 구성되며 $S = \langle tid, itemset, ClassLabel, timestamp \rangle$, $itemset \in I$로 표현한다. 항목집합은 분류 기준을 나타내는 항목들을 포함하며 클래스 라벨은 각 트랜잭션이 속한 클래스를 나타낸다.

정의 3.11 : 시간 분류규칙은 <ClassRule, TimeExp> 형태로 표현된다. ClassRule은 $\langle ClassModel \rangle$형식을 가진다. TimeExp는 시간표현식으로 $\langle ClassModel \rangle$이 소속된 시간 제약 조건을 명시한다.

예를 들면 고객의 신용도 예측을 위하여 의사결정트리 모델을 출력하는 분류 규칙을 <Decision_Tree, for (Years(1998), Years(2000))>라고 하면 이 규칙은 1998년~2000년까지만 해당하는 규칙이다.

(그림 1)과 같은 시간의 상하 관계를 나타내는 임의의 시간 계층을 *TH*라고 정의한다..

정의 3.12 : 시간 분류규칙 탐사는 트레이닝 데이타 집합 *TS*로부터, 시간 계층 *TH*를 이용하여, 사용자가 미리 지정한 시간제약 조건 TimeExp를 만족하는 규칙 <*ClassModel*, *TimeExp*>을 탐사하는 문제이다.

예를 들면 고객의 신용도 예측을 위하여 고객의 나이를 기준으로 사용할 때 *TH*를 이용하여 고객 그룹을 일반화함으로써 보다 함축된 분류 규칙을 생성할 수 있다.

시간 지식탐사에 있어 타임스탬프를 포함한 사건 또는 트랜잭션 데이터베이스는 [29]에서 제시된 바와 같이 시간 간격 일반화를 통해 시간 간격 데이터베이스로 변환된다. 이렇게 함으로써 서로 다른 시점에 발생된 사건들을 시간 간격을 갖는 사건시퀀스로 변환하거나 입력 데이터베이스를 효과적으로 요약한 데이터베이스를 얻으며 시간간격 연산자를 적용함으로써 유용한 시간 지식을 탐사한다. [46]는 시간간격 일반화에 관하여 자세히 언급하고 있다.

그 외에도 시간 특성화, 시퀀스 마이닝 문제도 유사한 방법으로 정의할 수 있으나 이 논문에서는 생략한다.

3.2 시간 데이타마이닝 질의어

문제 정의에서 명시한 시간 지식을 탐사할 수 있는 TMQL 문법을 정의한다. TMQL은 관계형 데이타베이스를 대상으로 기존의 DMQL을 시간지식 탐사를 수행할 수 있도록 확장하였다[19]. TMQL은 지식 탐사를 위한 KDL(Knowledge Discovery Language)과 지식 명세를 위한 KSL(Knowledge Specification Language)로 구성된다. 본 논문에서는 DMQL을 기반으로 하여 확장된 TMQL을 다음과 같이 확장된 부분에 한하여 제시한다.

3.2.1 지식 탐사 문법

KDL은 다음과 같이 BNF로 정의할 수 있다. KDL의 키워드는 이텔릭체로 표현된다.

```
<KDL> ::=
        mine                    <temporal_pattern_spec>
        related to              <attr_list>
        from                    <source_relation(s)>
        where                   < temporal_predicate(s)>
        with thresholds         <threshold_expression(s)>

<temporal_pattern_spec> ::= association rules
          | classification rules
            according to        <attr_list>
          | characterization rules
            according to        <attr_list>
          | sequential pattern
```

<temporal_pattern_spec>은 탐사할 시간 지식의 종류를 명시하는 절로서 종류마다 다른 형태를 가진다. <attr_list>는 연관규칙 탐사, 분류 등에서 사용하는 릴레이션 내의 항목명이고 <source_relation(s)>는 지식 탐사를 위한 시간 데이타가 저장된 릴레이션들을 명시한다. <temporal_predicate(s)>은 시간 지식의 시간 패턴을 명시하는 절로서 정의 3.5에서 명시한 시간 표현식으로 구성된다. <threshold_expression(s)>은 시간 지식의 유용성을 평가하기 위한 임계치를 주는 절로서 지식 종류마다 적합한 여러 종류의 임계치를 줄 수 있다. 이러한 <temporal_pattern_spec>에 대한 자세한 탐사 예는 아래와 같다.

다음은 KDL을 사용하여 시간 지식을 탐사하는 예를 보인 것이다. 쇼핑몰 데이타를 구성하는 세개의 릴레이션이 존재한다고 가정한다.

```
Customer(name, age, sex, salary, status, birth_date, birth_place,
        address, credit)
Items(item_no, item_name, brand, category, retail_price)
Purchase(transaction_no, transaction_time, customer, item_no, qty,
        amount)
```

● 시간 연관 규칙 탐사 예

1997년 1월부터 2000년 12월까지, 매년 가을철에 고객이 동시에 구매하는 경향이 높은 상품의 연관성을 탐사하는 경우에 다음과 같이 질의한다.

```
mine            association rules
related to      item_no
from            Items
where           for(Years(1997)-Months(1),
                Years(2000)-Months(12))
and             periodicity(Years-Months(9 : 10))
with thresholds support = 0.6, confidence = 0.75,
                frequency = 0.8
```

where 절에서 for 문은 정의 3.5의 시간표현식에서 캘린더표현식으로 구성된 기간 CI^+,를 명시하고 periodicity 문은 주기표현식 PT^+를 명시한다.

● 시간 분류 규칙 탐사 예

1998년~2000년 사이에 가입한 고객의 나이, 성별, 급여, 구매일에 의해 신용등급을 분류하기 위한 규칙을 탐사하는 질의는 다음과 같다.

```
mine            classification rules
                according to credit
related to      age, sex, salary, transaction_time
from            Customers, Purchase
where           for (Years(1998)-Months(1),
                Years(2000)-Months(12))
```

according to절은 credit라는 이름을 가진 정의 3.12의 <*ClassModel*>을 명시하고 where 절은 시간 제약 조건 <*Time Exp*>를 명시한다.

● 시간 특성화 규칙 탐사 예

고객의 나이, 성별, 급여, 출생지, 주소에 의해 전체 고객을 요약하여 고객 그룹의 특성을 탐사하기 위한 특성 규칙을 탐사하는 질의는 다음과 같다.

```
mine            characterization rules
                according to credit
related to      age, sex, salary, birth_place, address,
                count(*)%
from            Customer
with thresholds noise = 0.5
```

count(*)%는 각 요약된 고객 그룹별로 그룹에 속한 고객수를 말하고 noise = 0.5는 특성화 시에 발생할 수 있는 오차에 대한 허용율이다. 또한 요약시에 사용하는 시간 계층은 사용자가 임의로 지정할 수 있고 상기와 같이 생략시에는 사전에 지정된 시간 계층을 사용한다.

● 주기적 순차패턴 탐사 예

매년 여름철에 고객이 순차적으로 구매하는 경향이 80%인 상품의 시퀀스를 탐사하는 경우에 다음과 같이 질의한다.

```
mine            sequential pattern
related to      A.item_no, B.transaction_time
from            Items A, Purchase B
where           periodicity(Years-Months(6 : 8))
with thresholds support = 0.8
```

3.2.2 지식 명세 문법

KSL은 다음과 같이 BNF로 정의할 수 있다.

```
<KSL> ::=
    define      <knowledge_spec>
    for         <attr_name>
    as          <knowledge_expression>
```

<knowledge_spec>은 정의할 지식의 종류와 이름을 명시하는 절이다. <attr_name>는 지식 탐사 과정에서 지식이 적용될 항목이고 <knowledge_expression>은 지식을 명시한 절이다.

● 시간 계층 선언 예

구매 트랜잭션의 발생시간 항목에 대한 시간 계층을 명시하는 명령은 다음과 같다.

```
define      time_hierarchy purchase_time
for         <transaction_time>
as          {saturday, sunday, national holiday}
            < {holiday} < month < year
```

● 캘린더 지식 선언 예

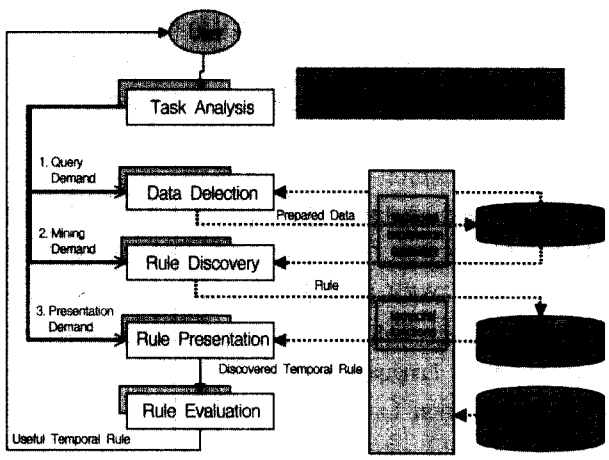
공휴일에 대한 캘린더 지식을 명시하는 명령은 다음과 같다. *는 특정 항목이 아닌 모든 항목에 캘린더 지식이 적용되는 것을 의미한다.

```
define      calendar national_holiday
for         *
as          periodicity(Months(1)Days(1),
            Months(4)Days(5),Months(12)Days(25))
```

그 외에도 TMQL 문장을 이용한 다양한 지식 탐사와 지식 정의가 가능하다.

3.3 지식 탐사 모델

시간 데이타마이닝 시스템을 설계하기에 앞서 시간 지식 탐사를 위한 처리 흐름을 모델링한다. 탐사 모델은 (그림 2)와같이 크게 작업분석(task analysis), 데이타 액세스(data access), 규칙 탐사(rule discovery), 규칙 출력(rule presentataion), 시간 지원(time support) 프로세스로 구성되며 원하는 규칙이 탐사될 때까지 지식 탐사를 반복한다.



(그림 2) 시간 지식 탐사 모델

● 작업 분석

사용자가 지식 탐사를 위해 입력한 TMQL 질의문의 문

법과 의미를 분석하여 마이닝 작업을 파악하는 프로세스이다. TMQL 질의에 내포된 시간표현식을 해석하기 위해 (그림 2)의 시간 지원 프로세스를 호출한다. 해석이 끝나면 데이타 액세스, 규칙 탐사, 규칙 표현 모듈을 차례로 호출하여 지식 탐사를 제어하는 프로세스이다.

- 데이타 액세스

소스 데이타베이스를 검색하여 TMQL 질의문에서 명시한 조건을 만족하는 데이타(테이블, 항목)을 추출한 후, 규칙 탐사 프로세스에서 원하는 형태로 가공하여 저장한다. TMQL 질의문의 시간 제약 조건을 비교하기 위하여 시간 지원 프로세스를 호출한다.

- 규칙 탐사

TMQL 질의문에서 명시한 데이타마이닝 요구에 따라 해당 알고리즘을 실행하여 데이타액세스 프로세스에 의해 저장된 데이타로부터 규칙을 탐사한다. 데이타마이닝 요구는 지식의 종류와 지식이 만족해야 할 임계치이다. 탐사된 규칙은 지식 베이스에 저장된다.

- 규칙 출력

탐사된 규칙을 사용자가 원하는 형태로 가공하여 출력한다. 출력 형태는 테이블, 그래프, 3차원 그래프 등이다.

- 시간 지원

시간 지식 탐사를 위해 중요한 프로세스로서 다른 모듈에게 시간 지원 기능을 제공한다. 이러한 시간지원 기능은 기존의 데이터 마이닝 각 프로세스에 대하여 시간 표현식 평가기를 통하여 다양한 캘린더 지식에 대한 처리를 수행할 수 있도록 다양한 시간 함수들을 구성하여 처리한다. 특히 3장의 TMQL에서 명시된 시간 표현식을 해석, 평가하고 실행하기 위하여 시간지원 모듈상의 시간표현 평가기를 호출하여 처리한다. 캘린더 지식을 이용하여 시간표현식을 평가하거나 지식 명세를 위한 KSL 명령문을 실행하여 캘린더 지식으로 저장한다.

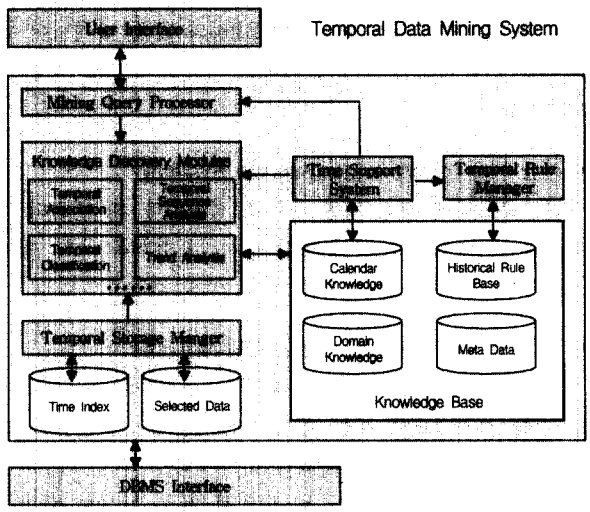
사용자는 출력된 규칙을 분석하여 원하는 유용한 규칙인지 평가한다. 원하는 규칙이면 중지하고 아니면 원하는 규칙이 탐사될 때까지 임계치 등을 바꾸어 가며 탐사 작업을 반복한다.

4. 시간 데이타마이닝 시스템 구조

3장의 시간 데이타마이닝 모델에 의거하여, 사용자가 입력한 TMQL 질의를 해석하여 시간 지식을 탐사 할 수 있는 시간 데이타마이닝 시스템의 구조는 (그림 3)과 같다. (그림 3)은 다음과 같은 여러 모듈로 구성된다.

- 마이닝 질의 처리기(Mining Query Processor)

사용자가 입력한 TMQL 질의문의 문법과 의미를 분석하여 마이닝 작업을 수행하고 마이닝 결과를 전달한다. 먼저 TMQL에서 명시한 시간표현식을 해석하기 위하여 시간지원 시스템을 호출하고 해석 결과에 따라 최적화된 지식 탐사 실행 계획(knowledge discovery execution plan)을 생성한 후, 규칙 종류에 따라 해당 지식탐사 모듈을 호출한다.



(그림 3) 시간 데이타마이닝 시스템 구조

- 시간지식 탐사 모듈(Knowledge Discovery Modules)

시간 연관규칙 탐사, 시간 분류, 시간 시퀀스 분석, 경향 분석 등과 같은 시간 데이타마이닝 기법 별로 알고리즘을 구현한 모듈이다. 사용자가 TMQL에서 명시한 입력 데이타를 시간 저장 관리자로부터 전달 받아 최소 임계치 이상을 만족하는 시간 지식을 탐사한 후, 결과를 지식 베이스에 저장하고 마이닝 질의 처리기에게 전달한다.

- 시간 저장 관리자(Temporal Storage Manager)

시간 데이타마이닝 알고리즘은 데이타의 증가와 반복적인 데이타 검색으로 인해 탐사 비용이 증가된다. 그러므로 대용량 시간 데이타를 효율적으로 저장하고 검색할 수 있는 시간 저장 관리자가 필요하다.

시간 저장 관리자는 DBMS인터페이스를 통해 필요한 데이타를 추출하여 데이타 화일로 로드 한다. 시간 색인(time index)은 데이타 화일과 규칙 베이스에 저장된 시간계층과 같은 도메인 지식을 이용하여 B 트리와 유사한 계층 형태로 구성되어 리프(leaf) 노드는 최소 시간단위를 가진다. 시간 색인의 초기 생성 비용은 소요되나 반복적인 데이타 검색 시에 성능 향상을 기대할 수 있다.

- 지식 베이스(Knowledge Base)

시간 마이닝을 위해 필요한 모든 지식과 지식탐사 모듈에 의해 탐사된 시간 규칙을 저장하는 저장소로 캘린더 지식, 도메인 지식, 이력 규칙 베이스(historical rule base), 메타 데이타(meta data)로 구성된다. 캘린더 지식은 TMQL에

서 명시한 시간표현식을 해석하기 위한 지식으로 요일, 주, 월, 공휴일 등으로 구성된 캘린더를 정의한다. 도메인 지식은 시간 일반화를 위한 시간 계층 등을 포함한다. 이력 규칙 베이스는 이력 정보를 가지는 시간 규칙을 저장한다. 예를 들면 시간 연관규칙의 경우 동일한 연관규칙이더라도 지지도와 신뢰도는 시간에 따라 변화하므로 지지도와 신뢰도의 이력을 같이 관리함으로써 2.2절의 규칙 마이닝이 가능하다. 메타 데이타는 지식 베이스 자체의 구조에 대한 정보를 가진다.

- 시간 지원 시스템(Time Support System)

다른 모듈에게 시간 지원 기능을 제공한다. 즉 캘린더 지식을 이용하여 TMQL에서 명시된 시간표현식을 해석, 평가하고 시간 함수들을 제공한다.

- 시간 규칙 관리자(Temporal Rule Manager)

2.3절에서 언급한 점진적 마이닝을 수행하여 이력 규칙 베이스를 정기적으로 갱신, 관리한다. 즉, 데이타 갱신이 발생할때 갱신된 데이타에 대해서만 마이닝을 수행하고 이력 규칙 베이스에 저장된 기존의 규칙과 새로 탐사한 규칙을 비교하여 규칙을 추가, 변경, 삭제한다.

- 사용자 인터페이스(User Interface)

시간 데이타마이닝 시스템을 보다 편리하게 사용할 수 있도록 TMQL을 모르는 사용자가지식 탐사를 수행할 수 있고 지식 탐사 결과를 2차원, 3차원 형태로 축약하여 보여주는 비쥬얼라이제이션 도구를 제공한다.

5. 시스템 구현 방안 및 적용

5.1 구현 방안

기존 데이타마이닝 시스템들은 DBMS와의 결합 방식에 따라 크게 독립 구조(stand-alone architecture)와 내장 구조(built-in architecture)로 분류할 수 있다.

독립 구조는 DBMS와 연결하여 데이타마이닝 기능을 수행하는 별도의 데이타마이닝 도구를 구현한 방식이다. 이 방식은 기존 DBMS 상에 데이타마이닝 엔진을 별도로 구현하는 전위처리(front-end) 방식으로서 DBMS와 독립적으로 운영되므로 느슨한 결합(loosely coupled) 방식이라고도 한다. 데이타마이닝 도구는 DB 인터페이스를 통해 DBMS에게 질의를 보내어 데이타베이스 또는 데이타웨어하우스로부터 데이타를 추출, 가공한 후 데이타 화일로 로드한다. 그리고 데이타 화일을 대상으로 데이타마이닝 알고리즘을 실행하여 그 결과를 레포트나 그래프 차트 형태로 보여 준다.

이 방식의 장점은 기존 데이타마이닝 알고리즘이 데이타 화일을 대상으로 개발되었다는 점에서 기존 DBMS를 변경하지 않고 저렴한 비용으로 데이타마이닝 기능을 구현할 수 있다는 점이며, 단점은 데이타마이닝 도구와 DBMS 간의 통신 부하로 인한 성능의 저하, DBMS 내부 모듈과의 연결이 이루어지지 않아 트랜잭션 관리, 권한 검사 등이 어렵고, 응용 프로그램 개발을 위한 표준 API를 제공하지 못하는 점이다. SAS Enterprise Miner, IBM Intelligent Miner 등과 같은 현존하는 대부분의 제품이 이 방식에 속하며, 앞서 언급한 기존 시간 마이닝 시스템들도 이 방식에 속한다[22].

내장 구조는 데이타마이닝 엔진이 DBMS의 구성 모듈로서 DBMS 내부에 내장, 통합되는 방식이며 밀 결합(tightly coupled) 방식이라고도 한다. 이 방식은 데이타마이닝 규칙을 저장, 관리하기 위한 규칙 베이스의 생성, 데이타마이닝 규칙의 추가, 갱신, 삭제, 접근 제어 등이 DBMS 내부에서 처리된다. 예를 들면 RDBMS에서 데이타마이닝 규칙 베이스는 SQL DDL문에 의해 테이블로 선언되고, 데이타마이닝 작업에 의해 생성된 규칙은 INSERT 문에 의해 테이블로 추가된다. 이 방식은 DBMS와 통합되었다는 점에서 독립 구조보다는 발전된 방식으로 MS SQL Server 2000에서 제공하는 데이타마이닝 엔진이 이 부류에 속한다[1].

이 방식의 장점은 모든 데이타마이닝 작업이 DBMS 내부에서 수행되므로 성능이 독립 구조 보다 우수하고 API를 제공하여 확장성이 뛰어나다는 점이며, 단점은 기존 DBMS에 데이타마이닝 기능을 확장하여 새롭게 변경해야 하므로 개발 비용이 많이 든다는 점이다.

이 두 방식은 서로 장단점이 있으나 성능 문제와 DBMS의 기능 확장 추세 등으로 인해 독립 구조에서 내장 구조로 발전되고 있는 추세이다. (그림 3)의 시간 데이타마이닝 시스템은 독립구조 또는 내장구조로 모두 구현이 가능하다.

(그림 3)을 RDBMS 상에서 독립구조로 구현할 경우, TMQL 질의 Q가 주어졌을때 Q를 처리하기 위한 전체적인 알고리즘은 다음과 같다.

① 질의 해석 단계 : 마이닝질의처리기는 RDBMS의 질의처리와 유사하게 Q를 파싱하여 질의 트리를 생성한다. 이때 Q의 where절에서 명시한 시간표현식을 해석하기 위하여 시간지원 시스템을 호출한다. 예를 들어 시간지원 시스템은 공휴일이라는 시간표현식을 해석하기 위하여 지식베이스에 저장된 캘린더 지식을 이용하여 시간 간격의 집합인 캘린더를 생성한다.

② 질의 최적화 단계 : 질의 트리로부터 최적화된 지식탐사 실행 계획을 생성한다. 지식탐사 실행계획은 실행 가능한 관계 대수(relational algebra)의 집합인 RDBMS의 실행계획과 달리, 시간지식 탐사 모듈이 필요로 하는 입력 파라미터이다. 입력 파라미터는 호출해야 할 시간 마이닝 알고리즘(예. 시간 연관규칙), 입력 데이타(소스 릴레이션 및 항목)를 추출하기 위한 SQL 질의문의 집합, 캘린더, 임계치(최소 지지도, 신뢰도, 발생도) 등으로 구

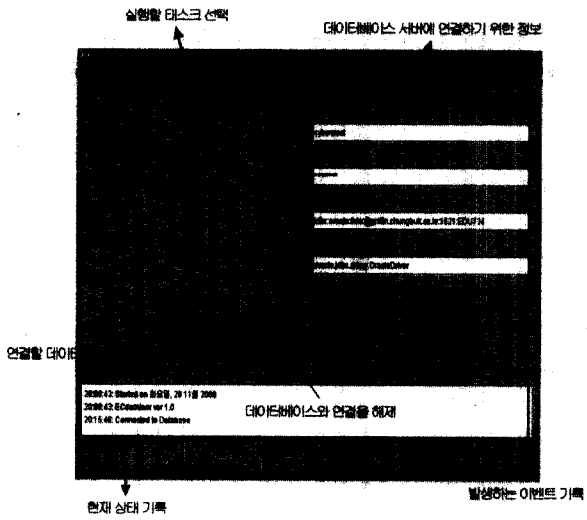
성된다. 이 단계에서의 최적화 목적은 각 시간 마이닝 알고리즘의 특성에 따라 가장 적은 비용이 드는 최소의 질의문을 생성하는 것이다. 예를 들어 시간 특성화 알고리즘에서는 요약을 위하여 sum, count, avg 등과 같은 SQL집계 함수(aggregate function)를 많이 사용하므로 최소의 집계 비용이 소요되는 질의문을 생성한다.

③ 질의 처리 단계 : 지식탐사 실행계획에 따라 지식탐사 작업을 수행한다. 먼저 시간저장 관리자는 실행계획의 SQL 질의문을 DBMS 인터페이스로 전달하여 데이타베이스로부터 필요한 데이타를 추출하여 데이타 화일로 로드한 후, 시간 색인을 구성한다. 지식탐사 모듈은 데이타 화일과 시간색인으로부터 시간지식을 탐사한 후, 그 결과를 지식베이스에 저장하고 마이닝질의처리기에게 전달한다.

5.2 EC-DaMiner의 확장 구현에의 적용 및 평가

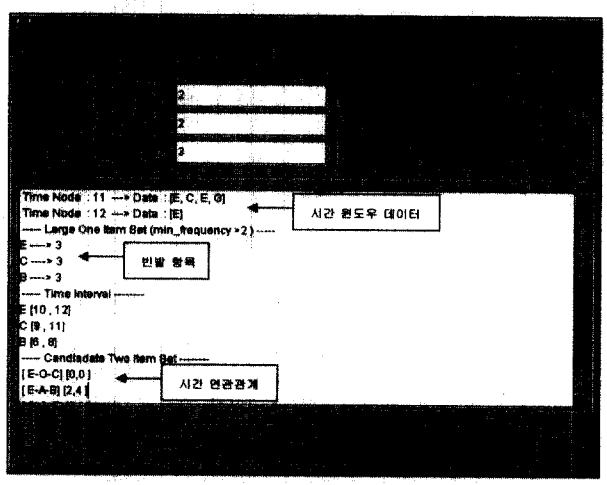
5.2.1 시간 데이터마이닝 프레임 워크의 적용

시간 마이닝 연산을 지원하는 마이닝 시스템은 [47]연구에서 설계 구현되었던 마이닝 시스템인 EC-DaMiner를 확장한다. EC-DaMiner는 e-Business시스템에 통합되어 비즈니스 태스크에 적용가능한 규칙을 생성하여 제공하는 시스템으로써 Java언어를 통해 구현되었으며 데이터베이스와는 JDBC인터페이스를 이용하는 독립구조 시스템이다. 이 시스템의 특징은 확장성과 통합성으로 마이닝 연산을 시스템에 확장하기 유연하며 기존의 데이터베이스 시스템과 통합을 지원하고 있다. 또한 데이터마이닝 질의어로써 MQL(Mining Query Language)가 제공되고 있다. 4장에서 제시된 시간 데이터 마이닝 시스템은 시간 마이닝 연산자 수준과 시스템 수준에서 구현된다.

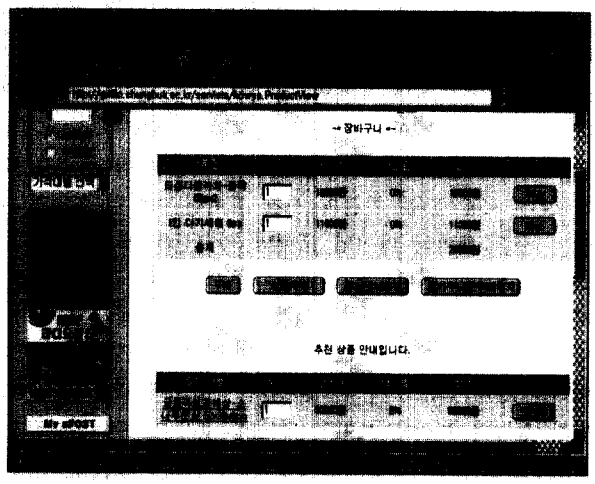


(그림 4) EC-DaMiner 시스템 UI

(그림 4)와 같이 구현된 EC-DaMiner시스템과는 달리 시간 데이터 마이닝 시스템은 시스템 수준에서 시간 저장 관리자 및 시간 지원 시스템과 시간 규칙 관리자의 세 가지 내부적 컴포넌트로의 확장이 필요하며 연산자 수준에서 시간연관규칙탐사, 시간 순서 패턴 탐사, 시간 분류 등이 포함되어야 한다. 현재는 시스템 수준에 있어 시간 데이터마이닝 질의어 형식인 TMQL을 지원하기 위한 마이닝 질의어 처리기의 확장구현과 시간 간격데이터 규칙 탐사 연산자가 일부 구현되었다. (그림 5)는 시간 간격데이터로부터 시간 연관관계를 추출하는 연산자를 구현한 시간데이터 마이닝 시스템의 일부를 보여준다.



(그림 5) 시간 데이터마이닝 연산자 구현



(그림 6) EC-DaMiner를 통한 연관상품 추천

(그림 6)은 기존의EC-DaMiner를 이용하여 탐사된 규칙을 전자상거래 머천트 시스템에 적용한 예이다. 하지만 Intelligent Miner나 EC-DaMiner와 같이 기존의 일반적인 데이터 처리를 위한 마이닝 시스템들은 실 시스템에 적용 시 시간 의미를 포함한 규칙의 적용이 불가능하다. 따라서 시간 데이터마이닝 프레임워크 기반의 시간 데이터마이닝 시스템은 이러한 다양한 시간 의미와 관계에 기반한 지식을 탐사하고 적용할 수 있다.

5.2.2 평 가

〈표 3〉 기존의 시간 데이터 마이닝과 시간 마이닝 프레임워크와의 비교

	기존 시간 데이터 마이닝	시간 마이닝 프레임워크
시간 규칙	• 순차 패턴 탐사 • 유사 시계열 탐사 • 경향 분석,예외분석	• 시간 특성화 규칙 탐사,시간 분류 규칙 탐사 • 시간 연관 규칙 탐사, 시간 관계 규칙 • 규칙 마이닝, 점진적 시간 지식 탐사
마이닝 모델	• 트랜잭션 데이터 모델 기반 • 시점 데이터처리 중심 • 개별적 마이닝 알고리즘 수준 • 다양한 시간 모델에 관계 • 시간 제약 표현 가능	• 시간 데이터 모델 포함 • 시간 간격 데이터 고려 • 시간 데이터 연산자를 고려한 마이닝 연산 구현 제시 • 마이닝 질의어(TMQL) 제시 • 단일한 캘린더 시간 표현식 적용
장점	• 소량의 시간 데이터 연산 수행 비용 저렴 • 개별 마이닝 연산 구현이 용이	• 시간 의미, 관계 및 제약조건 탐사 가능 • 시스템적 구현 모델 제시 • 대용량 시간 데이터 처리 가능 • DBMS와의 통합 구조, 지식베이스 구성
문제점	• 시간 의미 및 관계 개념 미흡 • 시간 의미 및 관계 탐사 부족 • 소량의 시간 데이터에 적합 • 제한적인 시간 표현 및 상의한 표현 형식	• DBMS 내장 데이터마이닝 연산자의 지원 • 다중 시간 모델 지원 필요 • 데이터량에 따른 성능 개선 • 대용량 시간 데이터의 효율적 저장,검색 필요 • 시간 규칙의 유용성 평가
적용	• 전자상거래, 법률, 의료, 금융, 주식, 의사결정 지원 등	

제안된 시간 데이터마이닝 프레임워크의 시간 데이터 마이닝 모델에 기반한 시간 데이터마이닝 기법의 구현은 기존의 순차 데이터나 시계열 데이에 대한 데이터마이닝 기법과는 다른 시간의미 규칙을 탐사하는 것을 목적한다[43, 46]. 따라서 일반적인 마이닝 연산과의 성능 비교는 적합하지 않다. <표 2>는 기존의 시간 데이터 마이닝 연구와 제안된 시간 데이터 마이닝 프레임 워크간의 모델수준의 비교와 특징 및 문제점을 비교한다. (그림 5)와 같이 구현된 시간 간격데이터로부터의 시간 관계 규칙을 탐사하는 마이닝 기법은 기존의 Allen알고리즘에 비하여 요약된 시간 데이터의 경우 레코드 건수 및 시간 복잡도에 대하여 만족할 만한 성능의 개선을 보이고 있다[43, 46]. 이 마이닝 연산의 실험된 데이터로는 전자상거래 데이터를 사용하였기 때문에 시간 데이터마이닝 연산을 포함하는 시간 데이터 마이닝 시스템은 EC-DaMiner의 확장을 통해 구현될 수 있다. 그리고 효율적으로 시간 관계 규칙을 탐사하여 (그림 6)과 같이 기존의 시스템에 적용될 수 있다. 하지만 이 연구에서는 데이터 요약을 통한 성능 개선에도 불구하고 시간 데이터량의 증대에 따른 지속적인 성능 개선의 여지를 갖고 있으며 또한 환자 병력 및 웹 로그 등과 같이 여러 특성의 시간 데이터로부터의 시간 규칙을 탐사하기 위해서는 이러한 특성을 고려한 기법의 개발 및 다각적인 실험이 필요하다. 현재 구현된EC-DaMiner의 적용에 있어서도 시간의미를 포함하지 못하고 있지만 특정 시점 및 시간 간격 내의 데이터에 대한 좀더 의미 있는 시간 연관 규칙을 적용할 수 있는 시간 마이닝 시스템으로의 확장은 구현된 연산자를 통해 가능하며 이 시스템의 확장된 적용은 의사결정에 있어서의 많은 가치를 증대할 수 있다.

6. 결 론

시간 데이타마이닝에 대한 기존 연구는 유용한 시간 의미와 시간 관계를 탐사하는데 부족하며, 시간 데이타마이닝의 전체적인 측면보다는 시간 연관규칙 탐사와 같은 일부 기법만을 다루고 있어 이 분야에 대한 종합적이고 체계적인 연구가 이루어지지 않고 있다.

이 논문에서는 기존 연구의 문제점을 개선하기 위하여 시간 데이타마이닝에 대한 전체적인 프레임워크를 제시하였다. 먼저 기존 연구 내용을 분석하여 정리한 후, 이를 취합하여 형식화시킨 시간 데이타마이닝 모델을 정의하였다. 또한 시간 데이타마이닝 모델을 기반으로 시간 데이타마이닝 질의어를 설계하고 시간 데이타마이닝 시스템의 구조를 제안하였다. 또한 제안된 프레임워크에 기반하여 기존의 구현된 EC-DaMiner를 확장 구현함으로써 시간 마이닝 시스템의 유용성을 적용 사례를 통하여 제시하였다. 이러한 프레임워크는 시간 데이타마이닝에 대한 명확한 이해와 문제 파악에 도움을 줄 수 있다.

시간 데이타마이닝은 기존 데이타마이닝에 비해 시간 패턴을 포함한, 시간에 따라 변화하는 지식을 보다 정확히 탐사할 수 있으므로 의료, 법률, 금융 등의 다양한 분야에서 유용하게 활용할 수 있다. 그러나 기존의 많은 연구에도 불구하고 아직 해결되어야 할 많은 연구 주제가 있다. 특히 기존 데이타마이닝 기법에 시간 요소를 확장함으로 인해 시간을 고려한 새로운 개념과 기법에 대한 연구가 필요하다.

향후 연구 방향은 설계한 시간 데이타마이닝 시스템을 실제로 구현하여 다양한 시간 규칙의 유용성을 평가하는 일과 시간 지식의 효율적인 탐사를 위한 알고리즘 및 저장 구조, 시간 지식의 유용성 문제를 보다 심도 있게 연구하는 것이다.

참 고 문 헌

[1] 이정무, Introduction to Data Mining with SQL Server 2000, Microsoft Tech-Ed 2000, 2000.

[2] R. Agrawal, Tomasz Imielinski, and Arun Swami, "Database mining : A performance perspective," IEEE Transactions on Knowledge and Data Engineering, Vol.5, No.6, December, 1993.

[3] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," the VLDB Conference, Santiago, Chile, September, 1994.

[4] R. Agrawal, G. Psaila, E. Wimmers, M. Zaot, "Querying

shapes of histories," In Proc. Twenty-first International Conference on Very Large Databases, Zurich, Switzerland, 1995.

[5] R. Agrawal and R. Srikant, "Mining sequential patterns," In Proc. Eleventh International Conference on Data Engineering, 1995.

[6] R. Agrawal, King-Ip Lin, Harpreet S. Sawhney, and Kyuseok Shim, "Fast similarity search in the presence of noise, scaling, and translation in time series databases," the VLDB Conference, Zurich, Switzerland, Sept., 1995.

[7] J. M. Ale, G. H. Rossi, "An Approach to Discovering Temporal Association Rules," SAC'00, Italy, Mar., 2000.

[8] J. Allen, "Maintaining Knowledge about Temporal Intervals," Comm. Of the ACM, Vol.26, No.11, Nov., 1983.

[9] G. Berger and A. Tuzhilin, "Discovering unexpected patterns in temporal data using temporal logic," Temporal Databases - Research and Practice, Springer-Verlag, 1998.

[10] R. L. Blum, "Discovery, Conrmation, and Incorporation of Causal Relationships from a Large Time-Oriented Clinical Database : The RX Project," In Computers and Biomedical Research, 1982.

[11] S. Chakrabarti, S. Sarawagi, and B. Dom., "Mining surprising patterns using temporal description length," In Proc. Twenty-Fourth International Conference on Very Large databases, 1998.

[12] R. Chandra, Arie Segev, and M. Stonebraker, "Implementing Calendars and Temporal Rules in Next Generation Databases," ICDE '94, pp.264-273, Houston, Texas, 1994.

[13] X. Chen and I. Petrounias, "A framework for temporal data mining," In Proc. Ninth International Conference on Database and Expert Systems Applications, DEXA'98, 1998.

[14] X. Chen, I. Petrounias and H. Heathfield, "Discovering temporal association rules in temporal databases," In Proc. International Workshop on Issues and Applications of Database Technology, 1998.

[15] D. W. Cheung, Han, J., Ng, V. T. and Wong, C.Y., "Maintenance of discovered association rules in large databases : An incremental updating technique," ICDE '96, 1996.

[16] D. W. Cheung, S. D. Lee, and Kao, B., "A general incremental technique for maintaining discovered association rules," DASFAA '97, Melbourne, Australia, Apr., 1997.

[17] C. Faloutsos, M. Ranganathan, and Y. Manolopoulos, "Fast subsequence matching in time-series databases," In Proc. ACM SIGMOD Conference on the Management of Data, Minneapolis, USA, 1994.

[18] Minos N. Garofalakis, Rajeev Rastogi and Kyuseok Shim, "SPIRIT : Sequential Pattern Mining with Regular Expression Constraints," the VLDB Conference, Edinburgh, Scotland, UK, 1999.

[19] J. Han, Y. Fu, W. Wang, K. Koperski and O. Zaiane, "DMQL : A Data Mining Query Language for Relational Databases," Reserach Report, DB Research Laboratory, Simon Fraser University, 1994.

[20] J. Han, G. Dong, and Y. Yin, "Efficient Mining of Partial Periodic Patterns in Time Series Database," In Proc. Fifteenth International Conference on Data Engineering, Sydney, Australia, 1999.

[21] C. S. Jensen, et al, "A Consensus Glossary of Temporal Database Concepts," ACM SIGMOD Record, Vol.23, No.1, 1994.

[22] M. A. King, J. F. Elde V, "Evaluation of Fourteen Desktop Data Mining Tools," IEEE, 1998.

[23] J. Y. Lee, K. J. Oh, K. H. Ryu, "Integration with Spatiotemporal Relationship Operators in SQL," ACM-GIS, pp.165-167, 1998.

[24] H. Mannila and H. Toivonen, "Discovering generalized episodes using minimal occurrences," In Second International Conference on Knowledge Discovery and Data Mining (KDD'96), pp.146-151, 1996.

[25] H. Mannila, H. Toivonen, and A. I. Verkamo, "Discovery of frequent episodes in event sequences," Data Mining and Knowledge Discovery, 1(3), pp.259-289, November, 1997.

[26] K. W. Nam, D. H. Kim, K. H. Ryu, "The Spatiotemporal Relationship Operator," ITC-CSCC, pp.1035-1038, 1996.

[27] B. Ozden, S. Ramaswamy, and A. Silberschatz, "Cyclic association rules," Int'l Conference on Data Engineering, Orlando, 1998.

[28] C. Rainsford and J. F. Roddick, "Temporal data mining in information systems : a model," In Proc. Seventh Australasian Conference on Information Systems, 1996.

[29] C. Rainsford, Accommodating Temporal Semantics in Knowledge Discovery and Data Mining, PhD Thesis, University of South Australia, 1998.

[30] S. Ramaswamy, S. Mahajan and A. Silberschatz, "On the discovery of interesting patterns in association rules," the VLDB Conference, New York City, September 1998.

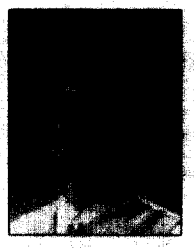
[31] S. Ramaswamy, Rajeev Rastogi and Kyuseok Shim, "Efficient algorithms for mining outliers from large data sets," the ACM SIGMOD Conference on Management of Data, Dallas, TX, May, 2000.

[32] J. F. Roddick, K. Hornsby and M. Spiliopoulou, "Temporal, Spatial and Spatio-Temporal Data Mining and Knowledge Discovery Research Bibliography," http : //www.cs.flinders.edu.au, 2000.

[33] J. F. Roddick and M. Spiliopoulou, "Temporal data mining : survey and issues," Research Report ACRC-99-007, University of South Australia, 1999.

[34] M. H. Saraee and B. Theodoulidis, "Knowledge discovery in temporal databases," In Proc. IEE Colloquium on Knowledge Discovery in Databases, 1995.

[35] Kyuseok Shim, R. Srikant and R. Agrawal, "High-dimensional similarity joins," the 13th International Conference on IEEE Data Engineering, 1997.
[36] Kyuseok Shim, Data Mining(Where are we heading for?), Data Mining Tutorial, 2000.
[37] A. Silberschatz and A. Tuzhilin, "What makes patterns interesting in knowledge discovery systems," IEEE Trans. on Knowledge and Data Eng., 8(6), pp.970-974, Dec., 1996.
[38] R. Snodgrass, "The Temporal Query Language TQuel," ACM TODS, Vol.12, No.2, Jun., 1987.
[39] R. Srikant and R. Agrawal, "Mining sequential patterns : generalisations and performance improvements," In Proc. International Conference on Extending Database Technology, Avignon, France, Springer-Verlag, 1996.
[40] T. Wade, D. Byrns, P. J., Steiner, J. F. and Bondy, J., "Finding temporal patterns- A set based approach. Articial Intelligence in Medicine," pp.263-271. 1994.
[41] S. Ye and J. A Keane, "Mining association rules in temporal databases," In Proc. International Conference on Systems, Man and Cybernetics, 1998.
[42] D. H. Kim, K. H. Ryu, H. S. Kim, "A Spatiotemporal database model and query language," The Journal of Systems and Software, Vol.5, 2000.
[43] J. S. Song, Y. J. Lee, and K. H. Ryu, "Discovering Temporal Relation Rules from Interval Data," submitted to the ETRI Journal, 2001.
[44] K. J. Jeong, Maintenance of Materialized View in Temporal Query Processing System, Ph.D Dissertation, Dept. of Computer Science, Chungbuk National University, 1998.
[45] S. N. Kim, Transforming Entity-Relationship into Object-Oriented Model in Temporal Paradigm, Ph.D Dissertation, Dept. of Computer Science, Chungbuk National University, 1997.
[46] Y. J. Lee, A Data Mining Technique for Discovering Temporal Relation Rules, Ph.D Dissertation, Dept. of Computer Science, Chungbuk National University, 1997.
[47] K. H. Ryu, J. W. Lee, Y. J. Lee, "Temporal Data Mining for eCRM," Database Research of Korean Information Science Society SIGDB, Vol.17, No.1, 2001.

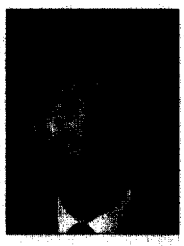


이 준 욱

e-mail : junux@dblab.chungbuk.ac.kr
1997년 충북대 컴퓨터과학과 졸업
1999년 충북대 대학원 전자계산학과(이학석사)
1998년~1999년 한국전자통신 연구원 위촉연구원 근무
1999년~현재 충북대학교 대학원 컴퓨터과학과 박사과정
관심분야 : 시간 데이터베이스, 시공간 데이터베이스, CRM, 시간 데이터 마이닝, 시공간 데이터 마이닝

이 용 준

e-mail : yjl@etri.re.kr
1984년 광운대학교 전산학과(학사).
1987년 연세대학교 전산학과(석사).
1993년 정보처리기술사(전자계산조직응용)
2001년 충북대학교 대학원 컴퓨터과학과(이학박사)
1984년~현재 한국전자통신연구원 우정기술연구부 책임연구원.
관심분야 : 시간 데이터베이스, 데이터 마이닝, 데이터베이스/네트 워크 보안



류 근 호

e-mail : khryu@dblab.chungbuk.ac.kr
1976년 숭실대 전산과 졸업
1980년 연세대학교 산업대학원 전산전공(공학석사)
1988년 연세대 대학원 전산전공(공학박사)
1976년~1986년 육군군수지원사전산실(ROTC 장교), 한국전자통신연구소(연구원), 한국방송통신대, 전산학과(조교수) 근무
1989년~1991년 Univ. of Arizona 연구원(TempIS Project)
1986년~현재 충북대학교 전기전자 및 컴퓨터공학부 교수
관심분야 : 시간 데이터베이스, 시공간 데이터베이스, Temporal GIS, 객체 및 지식베이스 시스템, 지식기반 정보검색 시스템, 데이터 마이닝, 데이터베이스 보안 및 Bio-Informatics