한국컴퓨터종합학술대회 2005 논문집 Vol. 32, No. 1(B)

수평 분산 데이터베이스 상의 세부 데이터 유출이 없는 순차 패턴 마이닝 기법

김승우[O] 원정임 박상현
연세대학교 컴퓨터과학과
{kimsw[O], jiwon, sanghyun}@cs.yonsei.ac.kr

Privacy Preserving Distributed Data Mining of Sequential Patterns on Horizontally Partitioned Databases

Seungwoo kim[0] JungIm Won, Sanghyun Park
Department of Computer Science, Yonsei University

요 약

본 논문에서는 수평 분산 데이터베이스에서 각 로컬 데이터베이스의 세부 데이터를 유출하지 않는 순차 패턴 마이닝 기법을 제안한다. 데이터 마이닝은 대용량 데이터베이스에서 유용한 지식을 추출하는 기법으로서 각광을 받고 있다. 그러나 분산 데이터베이스를 대상으로 마이닝을 수행하는 경우, 데이터 공유에 따른 개인 혹은 집단의 프라이버시가 유출될 수 있다는 문제점이 존재한다. 따라서 본 논문에서는 프라이버시 보호를 위하여 각 로컬 데이터베이스의 세부 데이터를 보호하면서도, 마이닝 결과의 정확성을 보장할 수 있는 새로운 순차 패턴 마이닝 기법을 제안한다. 제안된 기법에서는 우선, 세부 데이터의 유출을 방지하기 위하여 마이닝의 대상이 되는 항목과 항목간의 시간 선후 관계의 성립 여부를 벡터로 표현한 후, 이들 벡터간의 스칼라 프로덕트 연산을 수행하여 얻어진 결과를 패턴의 지지도로 활용하는 방안을 제안하였다. 또한, 연산 결과에 영향을 미치지 않는 벡터를 미리 제거하여 스칼라 프로덕트 연산에 따른 비용을 감소시키는 방안을 제안하였다.

1. 서론

최근 데이터 수집 및 저장 기술의 발달로 인하여 방대한 양의 데이터들이 축적되어 저장, 관리되고 있다. 이들 대용량 데이터들을 분석하여 얻어진 정보를 사용자 의사 결정에 활용하고자 하는 연구들이 최근 활발히 진행되고 있다[1].

데이터 마이닝이란 대량의 데이터로부터 쉽게 드러나지 않는 유용한 정보들을 추출하는 과정이다[1]. 즉, 사용 가능한 모든 데이터를 기반으로 숨겨진 지식, 기대하지 못했던 패턴, 새로운 법칙과 관계를 발견하여 이를 의사 결정 과정에 활용하는 것이다. 데이터 마이닝에는 연관 규칙(association rules), 순차 패턴(sequential patterns), 군집화(clustering), 분류(classification), 예측(prediction)과 같은 여러 가지 기법들이 널리 사용되고 있다. 이 중에서 순차 패턴은 연관 규칙에 시간 개념을 포함시킨 것으로, 임의의 시간 간격을 두고 순서대로 일어난 데이터를 분석하여 발생 빈도수가 높은 패턴을 찾아내는 기법이다[2]. 실세계에서 개개인에 의해 생성되는 데이터는 임의의 시간 간격을 두고, 여러 장소에서 발생되기 때문에 이질 형태를 갖는 분산 데이터베이스에 저장되는 경우가 빈번하다. 따라서 시간의 선후 관계를 기반으로 데이터 간의 연관성 혹은 규칙성 등을 분석하는 순차 패턴을 이용하여 이들 분산 데이터베이스 내의 데이터를 통합, 분석하는 것이 유용하다.

데이터 마이닝을 통해 얻어진 정보의 신뢰성과 유용성은 분석의 대상이 되는 원본 데이터의 크기와 질에 비례한다. 따라서 최근 여러 곳에 산재되어 저장, 관리되고 있는 분산 데이터베이스에서 대량의 양질 데이터를 수집하여, 이를 대상으로 데이터 마이닝을 수행하려는 시도들이 진행되고 있다[3][4].

분산 데이터베이스에서의 데이터 마이닝 적용 사례로 금융 분야를 들 수 있다. 여러 은행들이 저장, 관리하고 있는 기존의 고객 데이터 즉, 개인 신상, 거래 실적, 대출금, 상환 일자 등의 데이터를 데이터 마이닝을 통해 분석하여 고객에 대한 자금 회수 가능성 및 신용도 등의 정보를 파악하고, 이를 이용하여 신규 대출의 가능 여부 및 대출 금액의 한도 등을 조절함으로써 대출 시에 발생할 수 있는 위험도를 낮출 수 있다. 또 다른 사례로 여러 개의 소매점을 체인망 형태로 보유하고 있는 대형 마트들을 들 수 있다. 이들 마트들은 인근의 타 마트들로의 기존 고객의 이탈을 방지하고, 신규 고객을 유치하기 위하여 여러 곳에 분산되어 있는 데이터에 대한 데이터 마이닝을 통해 고객의 구매 패턴과 선호도, 고객의 그룹별 특성 등을 분석하여 신규 서비스 개발, 패키지 상품 개발, 수익성 높은 상품 입점 등의 집중적인 마케팅 전략을 세울 수 있다.

이렇게 분산 데이터베이스에서의 데이터 마이닝은 금융, 소매, 통신, 의료 등의 다양한 분야에서 폭 넓게 활용, 적용될 수 있다. 그러나, 기업 혹은 집단의 이익, 개인의 프라이버시 보호 등의 이유로 데이터를 소유한 집단이 데이터의 공유를 제한함으로 인하여 그 실용성에 한계가 있다. 제한적인 데이터 공유 환경에서의 마이닝 기법은 분류를 위한 의사 결정 트리를 생성하는 연구에서 시작되어[5], 연관 규칙[6][7], 클러스터링[3] 등의 연구로 확장되고 있다. 그러나 순차 패턴 마이닝 기법에 대해서는 아직 미흡한 상태이며, 이와 관련된 기존 연구 [8]에서 제안된 방법은 순차 패턴 마이닝의 대상이 되는 데이터베이스들에서 트랜잭션을 구성하는 항목이 동일하지 않은 환경, 즉 수직 분산 데이터베이스만을 대상으로 한다는 한계가 있다.

따라서, 본 연구에서는 분산 데이터베이스를 대상으로 세부 데이터의 유출을 방지하면서도, 마이닝 결과의 정확성을 보장할 수 있는 순차 패턴 마이닝 기법을 제안한다. 본 논문의 공헌은 다음과 같다. (1) 기존에 연구되지 않은 공통 항목을 갖는 수평 분산 데이터베이스를 데이터 마이닝의 대상으로 한다. (2) 데이터 공유 시, 세부 데이터의 유출을 방지하기 위한 데이터 보호 기법을 제안한다. (3) 일부의 공개된 데이터만을 이용한 순차 패턴 마이닝 기법을 제안한다. (4) 제안된 기법의 효율을 증진시키기 위한 개선 방안을 제안한다.

2. 관련 연구

분산 데이터베이스에서의 데이터 마이닝은 대량의 데이터를 대상으로 마이닝을 수행하여 보다 유용하고, 신뢰성 있는 정보를 추출하기 위하여 연구되어져 왔으며, 일반적으로 두 가지 방법이 있다. 첫 번째는 분산되어 있는 각 로컬 데이터베이스내의 모든 데이터들을 중앙의 한 데이터베이스에 저장해 놓은 후, 이를 데이터 마이닝하는 것이다[9]. 이 방법은 구현이 쉽고, 간단하지만 대량의 데이터를 중앙으로 전송하기 위한 통신 비용 및 저장 공간의 오버헤드가 크다는 단점이 있다. 두 번째는 각각의 로컬 데이터베이스를 대상으로 마이닝을 수행한 후, 그 결과를 집계하여 결과를 얻는 것이다[10]. 이 방법은 통신 비용 및 저장 공간에서 효율적이나, 데이터 마이닝을 통해 얻어진 결과의 정확성이 떨어진다는 단점이 있다.

데이터 마이닝에 있어서의 프라이버시 문제는 두 가지 측면에서 제기되었다. 첫 번째는 마이닝의 대상이 되는 데이터에 의해 개인의 사생활 정보가 유출되어 상업적인 용도로 도용되거나, 악의적인 사용자에 의하여 악용되는 것을 방지하는 것이며[11], 두 번째는 다수의 기업 혹은 집단의 데이터를 마이닝하는 경우, 각 집단의 비공개 데이터가 다른 집단으로 유출되는 것을 방지하는 것이다[5]. 분산 데이터베이스를 대상으로 각 로컬 데이터베이스내에 저장되어 있는 데이터들의 프라이버시를 보장하면서, 이들을 마이닝하기 위한 연구로 참고 문헌 [6][7]에서는 연관 규칙을 이용한 마이닝 기법을 제안하였으며, 참고 문헌 [4]에서는 나이브 베이즈 분류(Naive Bayes classification)을 이용한 마이닝 기법을 제안하였다. 또한, 참고 문헌 [3]에서는 군집화를 위한 마이닝 기법을 제안하였다.

특히, 참고 문헌 [6]에서는 로컬 데이터베이스 간에 공통 항목이 존재하는 수평 분산 데이터베이스를 대상으로 연관 규칙 마이닝 기법을 제안하였으며, 이를 위해 마이닝의 대상이 되는 각 데이터의 출처를 공개하지 않으면서도, 전체 데이터를 마이닝할 수 있는 상호 암호화(commutative encryption) 기법과 각각의 로컬 데이터베이스로부터 얻어진 지지도로부터 전체 지지도를 계산할 수 있는 시큐어 섬(secure sum) 기법을 제안하였다. 또한 참고 문헌 [7]에서는 공통 항목이 존재하지 않는 수직 분산 데이터베이스를 대상으로 하는 연관 규칙 마이닝 기법을 제안하였으며, 이를 위해 데이터베이스 내에서 각 항목(item)의 발생 여부를 벡터 형태로 표현한 후, 스칼라 프로덕트(scalar product) 연산을 수행하여 지지도를 계산하는 방법과

연산 시에 벡터 값이 공개되는 것을 방지하기 위한 보안 프로토콜을 제안하였다. 또한, 참고 문헌 [8]에서는 분산 데이터베이스를 위한 순차 패턴 마이닝 기법을 제안하였다. 각 로컬 데이터베이스 내에서 트랜잭션의 대상이 되는 데이터를 병합하여 시간 순서에 따라 오름차순으로 정렬한 다음, 정렬된 데이터를 발생 시간과 발생 여부로 분리하여 벡터 형태로 표현한 후, 스칼라 프로덕트 연산에 의하여 지지도를 계산하는 방법을 제안하였다. 이 때, 데이터의 유출을 방지하기 위하여 데이터를 매핑 테이블을 이용하여 일련의 숫자로 변환하였으며, 랜덤 방식에 의해 생성된 임의의 값을 데이터의 병합 과정에 추가하는 방식을 사용하였다. 그러나, 제안된 방법은 로컬 데이터베이스간에 공통 항목이 존재하지 않는 경우만을 대상으로 하며, 마이닝의 대상이 되는 데이터를 어느 로컬 데이터베이스에서 저장하고 있는지를 알 수 있으며, 마이닝에 의하여 생성된 패턴에 대하여 역 매핑을 수행하는 경우에 데이터의 유출을 방지하기 위하여 사용된 매핑 값을 유추해 낼 수 있으므로 프라이버시를 보장할 수 없다는 문제점이 있다.

본 논문에서는 기존에 연구되지 않은 공통 항목을 갖는 수평 분산 데이터베이스에서의 순차 패턴 마이닝 기법을 제안한다. 제안된 기법에서는 프라이버시를 보호하기 위하여는 참고 문헌 [6]에서 제안된 상호 암호화 기법과 시큐어 섬 기법을 사용하여 데이터의 출처를 감추고, 순차 패턴 마이닝에서의 지지도 계산을 위하여는 수직 분산 데이터베이스를 대상으로 제안된 [7]의 알고리즘을 확장, 적용한다. 수평 분산 데이터베이스의 경우, 각 로컬 데이터베이스내에 같은 값을 가지는 공통 항목이 출현할 수 있으므로, 참고 문헌 [8]에서와 같이 한 번의 스칼라 프로덕트 연산으로는 지지도를 구할 수 없다. 따라서, 본 연구에서는 스칼라 프로덕트 연산을 위한 알고리즘을 확장, 적용하는 방안을 제안하였으며, 연산을 위하여 생성된 벡터에서 연산 결과에 영향을 미치지 않는 부분을 미리 제거하여 연산 비용을 감소시키는 방안을 제안하였다.

3. 제안하는 기법

본 논문에서는 각 로컬 데이터베이스간에 공통 항목이 존재하는 수평 분산 데이터베이스를 대상으로 하며, 순차 패턴 마이닝 기법을 이용하여 로컬 데이터베이스내의 데이터들간의 관계, 규칙 등의 의미있는 유용한 정보를 찾아내고자 한다. 이를 위해 여러 개의 소매점을 체인망 형태로 보유하고 있는 대형 마트들을 예로 든다. 각 마트들은 <표 1>과 같이 고객 ID, 구매 시간, 구매 품목 등으로 구성된 구매 정보를 관리한다. 이때, 각 마트들간에는 같은 구매 품목들 즉, 공통 항목들이 존재한다. 이를 이용하여 고객의 구매 패턴 분석을 위한 마이닝을 수행할 경우, 지지도 계산을 위하여 고객 ID, 구매 품목 등은 반드시 공개되어야 한다. 그러나 각 마트는 마트의 수익을 위하여 해당 데이터가 공개되는 것을 제한할 수 있다. 따라서 본 논문에서는 제한적인 데이터 공유만을 허락하는 이들 데이터의 유출을 방지하면서도, 마이닝 결과의 정확성을 보장할 수 있는 순차 패턴 마이닝 기법을 제안한다.

3.1 수직 분산 데이터베이스를 위한 기법

참고 문헌 [8]에서는 각 로컬 데이터베이스간에 공통 항목이 존재하지 않는 수직 분산 데이터베이스를 대상으로 하는 순차 패턴 마이닝 기법을 제안하였다. 제안된 기법에서는 우선, 상세 데이터의 유출을 방지하기 위하여 각 로컬 데이터베이스내의 모든 항목을 매핑 테이블을 이용하여 유일한 값으로 매핑한다. 예를 들어 <표 1>의 마트 1에서의 {전자사전}은 1A, 마트 2의 {전기밥솥, 전자사전}은 2B, 마트 3의 {건전지}는 2C 등의 유일한 값으로 매핑한다. 다음, Apriori 알고리즘 [2]에 이용하여 매핑 값을 기반으로 후보 패턴 생성하고, 생성된 후보 패턴에 대한 지지도를 계산하여, 최소 지지도 이상을 만족하는 패턴만을 빈번 패턴으로 생성한다.

지지도 계산을 위하여 각 로컬 데이터베이스내의 모든 트랜잭션에 대한 발생 시간을 공유하며, 이를 이용하여 각 패턴의 실제 발생 유무를 표현하는 벡터와 패턴간의 시간 선후 관계가 성립하면 1, 성립되지 않으면 0의 값을 갖는 벡터를 생성한다. 생성된 벡터간의 스칼라 프로덕트 연산을 수행하여 얻어진 값이 해당 패턴의 지지도가 된다. 예를 들어, 매핑 값으로 변환된 패턴 {2A->2C}에 대한 지지도는 패턴 2A의 발생 시간 (20일 14시 25분, 20일 14시 30분, 26일 13시 40분[1])와 패턴 2C의 발생 시간 (21일 16시 25분[1], 27일 11시 30분, 24일 17시 20분[1])에 대하여 고객 ID별로 시간 값을 비교하여 2A의 발생 시간이 2C의 발생 시간 보다 이전인 경우 1, 아닌 경우 0으로 표현한 벡터 (1, 1, 0)를 구한다. 구해진 시간 벡터 (1, 1, 0)를 패턴 2A의 발생 여부를 표현한 벡터 (1, 1, 0)과 패턴 2C의 발생 여부를 표현한 벡터 (0, 1, 0)와 함께 스칼라 프로덕트 연산을 수행하여 패턴 {2A->2C}의 지지도로 1의 값을 얻는다.

<표 1> 각 마트의 고객 구매 정보

마트 1		
고객 ID	구매 시간	구매 품목
1	2월 20일 14시 25분	전기밥솥, 전자사전
2	2월 20일 14시 30분	전자사전
1	2월 26일 12시 20분	건전지, 쌀
마트 2		
고객 ID	구매 시간	구매 품목
3	2월 21일 16시 25분	전기밥솥, 전자사전
3	2월 23일 14시 25분	건전지
마트 3		
고객 ID	구매 시간	구매 품목
2	2월 22일 15시 30분	전기밥솥
2	2월 27일 11시 30분	건전지

3.2 수평 분산 데이터베이스를 위한 기법

3.1 절에서 제시된 기법을 본 논문에서 다루는 수평 분산 데이터베이스에 적용하는 경우 다음과 같은 문제점들이 발생한다. (1) 서로 다른 로컬 데이터베이스내에 동일 항목이 존재하는 경우, 이들 항목들은 매핑 테이블을 통해 서로 다른 매핑 값으로 변환된다. 따라서, 동일 항목이 서로 다르게 표현되므로 마이닝 결과의 정확성을 보장할 수 없다. (2) 임의의 항목에 대한 지지도 계산을 위하여 수직 분산 데이터베이스에서는 항목을 저장하고 있는 단 하나의 로컬 데이터베이스만을 엑세스하여, 항목에 대한 발생 빈도를 구하면 된다. 그러나, 수평 분산 데이터베이스에서는 임의의 항목이 여러 로컬 데이터베이스에 저장되어 있을 수 있으므로, 다수의 로컬 데이터베이스를 엑세스하여 항목에 대한 전체 발생 빈도를 구해야한다. 즉, 더 많은 데이터 전송 비용과 스칼라 프로덕트 연산에 대한 비용이 요구된다. (3) 매핑 값을 기반으로 생성된 빈번 패턴에 대하여 매핑 테이블을 이용한 역 매핑을 수행하는 경우, 실제 항목에 대한 정보가 유출될 수 있다.

<표 2> 마트 1, 2, 3의 구매 여부에 따른 표현

마트1								
고객 ID	전기밥솥		전자사전		쌀		건전지	
1	1	2월 20일 14시 25분	1	2월 20일 14시 25분	1	2월 26일 12시 20분	1	2월 26일 12시 20분
2	0	2월 24일 11시 30분[1]	1	2월 20일 14시 30분	0	2월 26일 13시 20분[1]	0	2월 24일 14시 50분[1]
3	0	2월 22일 16시 25분[1]	0	2월 26일 13시 40분[1]	0	2월 21일 12시 40분[1]	0	2월 23일 11시 10분[1]
마트2								
고객 ID	전기밥솥		전자사전		쌀		건전지	
1	0	2월 22일 19시 35분[1]	0	2월 24일 17시 45분[1]	0	2월 20일 13시 20분[1]	0	2월 24일 15시 40분[1]
2	0	2월 24일 17시 40분[1]	0	2월 26일 12시 00분[1]	0	2월 26일 18시 10분[1]	0	2월 23일 12시 20분[1]
3	1	2월 21일 16시 25분	1	2월 21일 16시 25분	0	2월 23일 14시 25분[1]	1	2월 23일 14시 25분
마트3								
고객 ID	전기밥솥		전자사전		쌀		건전지	
1	0	2월 27일 17시 45분[1]	0	2월 25일 19시 20분[1]	0	2월 22일 18시 40분[1]	0	2월 21일 16시 25분[1]
2	1	2월 22일 15시 30분	0	2월 23일 15시 00분[1]	0	2월 23일 15시 30분[1]	1	2월 27일 11시 30분
3	0	2월 24일 17시 40분[1]	0	2월 23일 17시 40분[1]	0	2월 26일 18시 30분[1]	0	2월 24일 17시 20분[1]

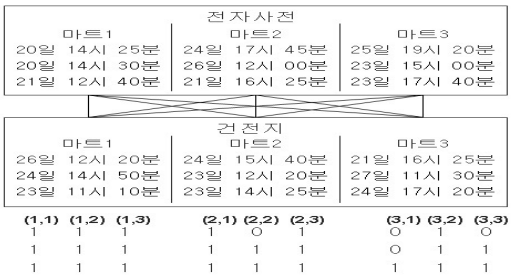
이러한 문제점 해결을 위하여 본 논문에서는 기존 [8]의 방식을 확장, 적용한 새로운 기법을 제안한다. 이를 위해 우선, 길이 1인 빈번 패턴 생성을 위하여 참고 문헌 [6]에서의 상호 암호화(commutative encryption) 기법과 시큐어 섬 기법을 사

1) 임의로 생성된 시간 데이터

용한다. 상호 암호화 기법은 적용 순서에 관계없이 암호화 기법을 중첩, 적용하여 데이터를 암호화하는 기법으로, 각 로컬 데이터베이스에 대한 정보는 공개하지 않으면서, 리스트 형태로 표현된 전체 데이터를 구하는 방법이다. 예를 들어, <표 1>의 전체 마트에서 특정 수 이상 구매된 품목을 얻기 위하여 우선, 상호 암호화 기법을 이용하여 길이 1인 품목들로 구성된 전체 품목 리스트 <전기밥솥, 쌀, 전자사전, 건전지>를 생성한다. 다음 시큐어 섬 기법을 이용하여 전체 품목 리스트내의 각 품목에 대한 지지도를 계산한다. 시큐어 섬 기법은 각 로컬 데이터베이스에서의 임의의 패턴이 갖는 발생 빈도에 랜덤 방식에 의하여 생성된 임의의 수를 더함으로써, 각 로컬 데이터베이스에서 패턴의 실제 발생 빈도를 공개하지 않는 방식이다.

다음, 길이 1인 빈번 패턴으로부터 길이 2인 후보 패턴을 생성하고, 지지도를 계산한다. 이를 위하여 참고 문헌 [8]에서와 같이 길이 1인 빈번 패턴에 대하여 <표 2>와 같이 벡터를 생성한 후, 이를 대상으로 지지도를 구한다. 여기서, 고객의 품목에 대한 구매 여부를 0과 1의 벡터 값으로 표현하고, 구매가 이루어지지 않은 구매 시간에는 랜덤 방식에 의하여 생성된 임의의 값을 저장한다. 그러나, 지적된 바와 같이 [8]에서 제안된 기법을 그대로 공통 항목을 갖는 수평 분산 데이터베이스에 적용할 수 없으므로, 본 논문에서는 기존의 방식을 확장, 적용한 새로운 기법을 제안한다.

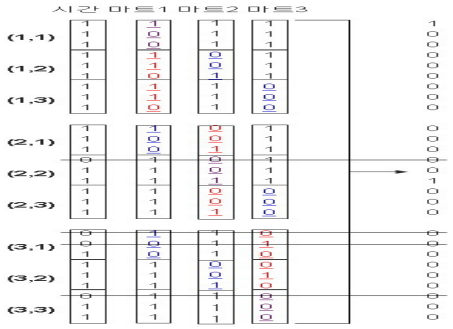
지지도 계산을 위하여 수직 분산 데이터베이스에서는 임의의 항목이 단 하나의 로컬 데이터베이스에만 존재하므로 한 번의 스칼라 프로덕트 연산을 수행한다. 그러나, 수평 분산 데이터베이스에서는 임의의 항목이 여러 로컬 데이터베이스에 존재하므로, 다수의 스칼라 프로덕트 연산을 수행하여야 한다. 예를 들어, 지지도를 구하고자하는 후보 패턴이 {전자사전->건전지}라면, 품목 '전자사전'에 대한 마트 1, 2, 3에서의 구매와 품목 '건전지'에 대한 마트 1, 2, 3에서의 구매를 모두 고려하여야 한다. 즉, 패턴이 생성될 수 있는 모든 경우를 고려하여 마트별 '전자사전'에 대한 3 벡터와 마트별 '건전지'에 대한 3 벡터를 조합하여 각각의 경우에 대해 스칼라 프로덕트 연산을 수행하고, 그 결과들을 합하여 해당 패턴의 지지도를 구한다. <그림 1>에 패턴 {전자사전->건전지}에 대하여 생성된 벡터를 보인다. 여기서 (1,3)은 마트 1에서 '전자사전'을 구매한 다음에 마트 3에서 '건전지'를 구매했는지의 여부를 각 고객별로 표현한 것으로, 구매 시간에 대한 선후 관계가 성립하면 1, 성립하지 않으면 0을 벡터 값으로 할당한다.



<그림 1> 패턴 {전자사전->건전지}에 대한 각 마트별 구매 시간 선후 관계를 표현하는 벡터

그러나 이 방법은 만약 로컬 데이터베이스의 수가 n이고, 패턴의 길이가 l이라면 총 n^l만큼의 연산을 필요로 하므로, 로컬 데이터베이스의 수와 패턴의 길이가 증가함에 따라 빈번 패턴을 생성하기 위한 스칼라 프로덕트의 연산 오버헤드가 급속히 증가하므로 비효율적이다. 따라서, 본 연구에서는 벡터를 로컬 데이터베이스별로 분리하여 여러 번에 걸쳐 처리하지 않고, 생성된 벡터를 순서대로 나열하여, 접합한 후, 한 번의 스칼라 프로덕트 연산으로 처리한다. 예를 들어, <그림 1>의 각 마트별 구매 시간의 선후 관계에 의하여 얻어진 벡터들과 각 마트에서의 구매 품목에 대한 구매 여부를 표현하는 벡터를 <그림 2>와 같이 마트 순서로 나열하여, 접합한다. 이 때, 마트 1의 경우에서와 같이 해당 마트가 관여되지 않은 벡터 (2,2), (2,3), (3,2), (3,3)은 계산 결과에 영향을 미치지 않기 위하여 모두 1로 구성된 벡터로 표현한다. 또한, (1,1), (2,2), (3,3)과 같이 서로 다른 품목이 한 마트에서 모두 구매된 경우에는 해당 품목들의 구매 여부 벡터를 서로 곱하여 하나의 벡터를 만든 후, 이 벡터를 나열한다. 이렇게 구성된 구매 시간에 대한 벡터들과 마트 1, 마트 2, 마트 3에서의 구매 여부에 대한 벡터들을 대상으로 스칼라 프로덕트 연산을 수행하여, 후보 패턴 {전자사전->건전지}에 대한 지지도로 1의 값을 연산 결과로 얻을 수 있다.

본 논문에서는 연산 비용을 보다 줄이기 위하여 시간 선후 관계를 표현하는 벡터를 마트 1, 마트 2, 마트 3에 전달하여 시간 벡터의 값이 0이 나오는 위치와 같은 위치에 있는 마트 1, 마트 2, 마트 3의 벡터 값을 모두 제거한다. 이는 시간의 선후 관계가 성립되지 않는 패턴을 미리 제거하여 전체 연산의 비용을 줄이기 위함이다. <그림 2>에서의 실선이 벡터에서 제거되는 부분으로, 이 부분을 제외한 나머지 벡터들에 대해서만 스칼라 프로덕트 연산을 수행하므로 연산 비용을 줄일 수 있다.



<그림 2> 패턴 {전자사전->건전지}에 대한 스칼라 프로덕트 연산 수행 과정

4. 결론

본 논문에서는 공통 항목이 존재하는 수평 분산 데이터베이스를 대상으로, 각 로컬 데이터베이스의 세부 정보를 유출하지 않으면서 마이닝 결과의 정확성을 보장할 수 있는 순차 패턴을 마이닝 기법을 제시하였다. 제안된 기법에서는 수직 분산 데이터베이스를 대상으로 제안된 [8]의 알고리즘을 확장, 적용하였으며, 알고리즘의 성능 향상을 위하여 각 로컬데이터베이스로부터 생성된 벡터들을 하나로 접합하여 연산하는 방식과 연산에 참여하지 않는 벡터들을 미리 제거하여 연산 비용을 감소시키는 방안을 제안하였다.

참고문헌

[1] J. Han and M. Kamber, "Data Mining : Concepts and techniques", Morgan Kaufmann Publishers, 2000.

[2] R. Agrawal and R. Srikant, "Mining sequential patterns," In Proceedings of the 11th International Conference on Data Engineering, 1994.

[3] S. Merugu and J. Ghosh, "Privacy-Preserving Distributed Clustering Using Generative Models," In Proceedings of the Third IEEE International Conference on Data Mining, 2003.

[4] M. Kantarcoglu and J. Vaidya, "Privacy Preserving Naive Bayes Classifier for Horizontally Pertitioned Data," In IEEE ICDM Workshop on Privacy Preserving Data Mining, 2003.

[5] Y. Lindell and B. Pinkas, "Privacy Preserving Data Mining," Proceedings of the 20th Annual International Cryptology Conference on Advances in Cryptology, August 2000.

[6] M. Kantarcioglu and C. Clifton, "Privacy-preserving distributed mining of association rules on horizontally partitioned data," In The ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery, 2002.

[7] J. Vaidya and C. Clifton, "Privacy preserving association rule mining in vertically partitioned data," In The Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2002.

[8] J. Zhan, L. Changy, and S. Matwinz, "Privacy-Preserving Collaborative Sequential Pattern Mining," Proceedings of Workshop on Link Analysis, Counter-terrorism and Privacy, 2004.

[9] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules," In Proceedings of the 20th Int'l Conference on Very Large Databases, 1994.

[10] L. Hall, N. Chawla, K. Bowyer, and W. Kegelmeyer, "Learning Rules from Distributed Data," Workshop on Large-Scale Parallel KDD Systems, 1999.

[11] R. Agrawal and R. Srikant, "Privacy-preserving data mining," Proceedings of the 2000 ACM SIGMOD international conference on Management of data, 2000.