USN 환경에서 의미 기반 트랜잭션 구조를 이용한 순차 패턴 탐사 기법

최필선, 강동현, 김환, 김대인, 황부현
전남대학교 전자컴퓨터공학과
e-mail:pilddong@nate.com

Sequence Pattern Mining Using Meaning-based Transaction Structure for USN system

Pilsun Choi, Donghyun Kang, Hwan Kim, Daein Kim, Buhyun Hwang
Dept of Electronic Computer Engineering, Chonnam University

요 약

순차 패턴 탐사 기법은 순서를 갖는 패턴들의 집합 중에 빈발하게 발생하는 패턴을 찾아내는 기법이다. USN 환경에서 발생하는 스트림 데이터는 시간 속성을 갖는 이벤트들의 집합으로 표현할 수 있으며 순차 패턴 탐사 기법을 이용하여 유용한 정보를 탐사할 수 있다. 그러나 스트림 데이터 환경에서는 데이터가 무한하고 연속적으로 발생하기 때문에 모든 데이터를 저장하여 패턴을 탐사하는 기법을 적용하는 데는 문제가 있다. 이 논문에서는 향상된 데이터 처리방식을 사용하여 순차패턴을 탐사하는 스트림 데이터 마이닝 기법에 대하여 제안한다. 제안하는 기법은 의미 단위의 가변적 윈도우를 사용하여 스트림 데이터로부터 트랜잭션을 생성하고 이 트랜잭션들의 집합을 해시와 슬라이딩 윈도우를 사용하여 스트림 데이터의 순차 패턴을 탐사한다. 이를 이용한 제안 기법은 실시간 시스템에 적합하게 데이터 저장 공간 사용의 효율성을 높이고 신속하게 유용한 패턴을 탐사할 수 있다.

1. 서론

현재 데이터 마이닝의 분야는 수많은 종류와 크기의 데이터를 분석하여 사용자의 요구사항에 맞게 유용한 정보를 추출할 수 있도록 많은 기법이 개발되고 있다. 그러므로 사용자의 요구가 많아짐에 따라 최대한 적은 비용으로 양질의 정보를 도출하는 방법에 대한 연구가 필요하다[1][2].

순차 패턴 탐사는 데이터 마이닝의 분야중에 하나로써 시간 속성을 갖는 아이템들의 집합에서 빈발하게 발생되는 패턴을 탐사하여 어떤 이벤트의 순차적인 발생 정보를 탐사하는 기법이다[3]. 순차 패턴 탐사의 대표적 알고리즘인 PrefixSpan 알고리즘[4]은 1항목 순차로부터 2항목 후보 순차들을 생성하고 이를 통해 다수 항목 후보 순차들의 빈발함을 계산하는 Apriori 성질을 이용하지만 후보 순차 패턴들의 집합을 생성하지 않고 패턴을 탐사하는 효율적인 알고리즘이다.

USN 시스템 환경에서 생성되는 데이터는 시간 속성을 가지고 있다. 하지만 이 시스템에서는 계속적으로 데이터가 생성되기 때문에 일정한 크기의 데이터를 분석하는 PrefixSpan 알고리즘은 유용하게 사용될 수 없다. 이와 같은 시스템을 보안하여 무한한 데이터가 끊임없이 생성되는 실시간 시스템에 적합한 알고리즘이 필요하다[5].

실시간 시스템에 적용하기 위해 개발된 알고리즘 중에 해시(Hash) 구조를 이용한 HAPT 알고리즘[6]이 연구되었다. 이 알고리즘은 빠른 검색 시간이 소요되고 예측가능한 패턴을 실시간으로 탐사한다는 장점이 있지만, 과거 자료를 모두 저장하기 때문에 과거 자료에 대한 처리 과정이 불분명하다. 이 논문에서는 USN 시스템 환경에서 발생되는 스트림 데이터를 이용한 HAPT 기법에 대하여 기술하고 이를 보안한 알고리즘을 제안한다.

이 논문의 구성은 다음과 같다. 2장에서는 USN 환경에서의 순차 패턴 탐사 기본원리와 이를 이용한 HAPT 알고리즘을 기술하고 기존 알고리즘에 대한 문제점을 논의한다. 3장에서는 순차패턴 탐사에 적합한 개선 알고리즘을 소개하고, 끝으로 4장에서는 결론 및 향후연구를 기술한다.

2. 관련연구

2.1 Apriori 와 PrefixSpan

데이터 마이닝의 중요한 원리인 Apriori 성질(Apriori Property)[7]은 데이터 마이닝의 탐색 효율성을 위한 중요 성질 중의 하나로 그 내용은 “빈발 항목 집합의 공집합이 아닌 모든 부분 집합은 반드시 빈발하다” 이다. Apriori 성질에 의하여 검색 비용을 줄여주는 원리는 다음과 같다. 특정 항목 집합이 빈발하게 발생하는 지를 탐사하는 경우, 그 항목이 발생된 트랜잭션의 수를 카운트하여 빈발함을 판단한다. 또한 빈발한 항목의 부분 집합 또한 빈발하다는 성질을 이용하여 큰 수준의 항목 집합의 빈발함에 대한 탐색 비용을 줄일 수가 있다.

순차 패턴 탐사는 실제적인 시간 변화에 따라 저장된

※ 이 논문은 2010년 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업 연구임.(2010-0005647)

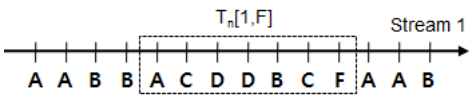
일련의 순서화된 요소 또는 사건으로 이루어진 순차 데이터베이스에서 공통적으로 빈발하게 발생하는 순차(요소, 사건, 패턴)를 탐사하는 데이터 마이닝 기법이다. 순차패턴 탐사에 대표적 알고리즘인 PrefixSpan 알고리즘[4]은 먼저 순차 데이터베이스를 검색하여 빈발하게 발생된 1항목 패턴을 탐사한다. 다음으로, 발견된 각 빈발 1항목 패턴을 이용하여 1항목 패턴에 대한 투영 데이터베이스를 생성한다. 투영 데이터베이스란, 발견된 각 빈발 항목을 각 순차에서 접두부(Prefix)로 정의하고 각 순차의 공통 접두부를 제외한 후두부(Suffix)를 나타낸 것이다. PrefixSpan 알고리즘의 성능은 투영 데이터베이스를 얼마나 생성하느냐에 따라 달라지며 각 트랜잭션의 순차 길이에 비례하여 탐사 시간이 소요된다.

2.2 HAPT 알고리즘과 문제점

PrefixSpan 알고리즘을 스트림 데이터 환경으로 확장한 기법 중에 해시(Hash) 구조를 이용한 HAPT 알고리즘[6]이 연구되었다. HAPT 알고리즘은 실시간 스트림 시스템에서 데이터의 양이 무한하게 증가하기 때문에 Apiori 성질을 이용할 수 없다는 점과 현재 빈발하지 않는 패턴이 미래에는 빈발한 패턴이 될 수 있다는 점을 보안한 알고리즘이다. 그리고 해시 구조를 사용하여 실시간 시스템에서 빠르게 예측 가능한 패턴을 찾기 위해 검색시간을 단축시킨 장점이 있다.

USN 환경은 어떤 현상에 대한 데이터를 획득하는 센서가 존재한다. 이 센서는 최대, 최소와 같은 범위를 가지고 있으며 발생하는 데이터의 범위를 예측할 수 있는데 이와 같은 데이터의 범위를 일정 단위로 나누어서 문자화 하는 것을 '특성화(Characterizing)'라고 한다. 예를 들어 습도를 나타낼 때에 퍼센트(%) 단위를 나타내는데 각 측정값마다 범위를 구분하여서 문자항목으로 변환할 수 있다. 0~30%는 A, 30~70%는 B, 70~100%는 C와 같은 문자로 나타낼 수 있는데 이를 USN 환경에서는 한 아이템으로 본다. 즉, 이와 같은 아이템들이 실시간으로 처리장치에 입력되는 시스템을 USN 시스템으로 정의하고 이 시스템에서 실시간으로 데이터 마이닝하는 기법이 HAPT 알고리즘이다.

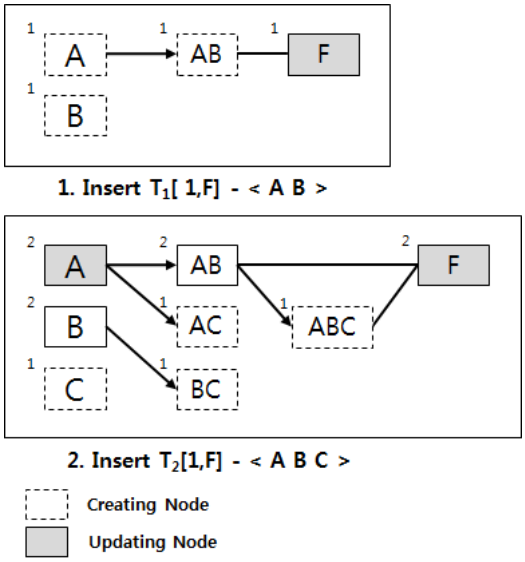
[6]에서는 USN 환경에서 트랜잭션을 생성할 때 '의미(Meaning)'있는 이벤트를 중심으로 그전에 발생했던 아이템들의 집합을 한 트랜잭션으로 정의하였다. 이 트랜잭션은 길이가 일정하지 않아 가변적으로 데이터를 수용할 수 있다는 장점이 있다.



(그림 1) 트랜잭션 생성

[6]에서는 트랜잭션을 생성할 때 시스템이 관심을 갖는 중요한 이벤트를 타켓 이벤트로 설정하고, 이 이벤트가 발생했을 때 타겟 이벤트를 기준으로 사용자가 정의한 길이 안에 발생했던 아이템들의 집합을 한 트랜잭션으로 생성한다. (그림 1) 에서는 타겟 이벤트를 F로 정의하고 F를 기준으로 6 만큼의 길이 전에 발생한 아이템의 집합을 한 트랜잭션으로 생성하였고 이는 Tn[1,F] : < A C D D B C > 로 표현할 수 있다. 여기서 n은 트랜잭션 번호이고 1은 스트림이 발생하는 센서의 고유번호이다.

생성된 하나의 트랜잭션은 그에 속해 있는 부분 순차를 찾아내서 이를 해시 구조를 이용한 HAPT 구조에 삽입하는데(Pattern Unfolding) 이 부분순차의 구조는 발생횟수와 이후에 발생하는 부분순차의 값으로 구성되어 있다.



(그림 2) HAPT 구조의 예

(그림 2) 는 2개의 트랜잭션 <AB>, <ABC>에 각각의 트랜잭션들의 부분 순차들을 해시 구조에 삽입하고 순서에 따라 각각 상위 순차에 해당하는 아이템셋(Superset)에 링크를 연결하는 HAPT 구조를 나타낸 것이다. 여기서 각각의 부분 순차에 해당하는 노드는 "발생횟수(Count)"와 상위순차 노드에 대한 연결정보를 가지고 있다. 예를 들어 <A> 노드에는 발생횟수가 2이며 <A> 이후에 발생한 상위순차 <AB>, <AC>에 대한 연결정보가 저장되어 있다. 이는 스트림 환경에서 어떠한 순차가 입력되면 그 순차에 대한 발생횟수와 상위순차에 대한 정보를 해시 구조를 이용해 빠르게 알 수 있기 때문에 예측가능한 패턴목록을 사용자에게 제공할 수 있다. 즉, 이 예측한 패턴목록을 바탕으로 사용자에게 더욱더 유용한 정보를 제공해 줄 수 있는 것이다.

이 HAPT 알고리즘은 USN 환경에서는 입력되는 값이 최대/최소 측정치와 같은 측정 범위를 갖기 때문에 발생하는 데이터의 범위를 알 수 있다는 감독된 시스템(Supervised System)이라는 특징을 이용하여 스트림 데이

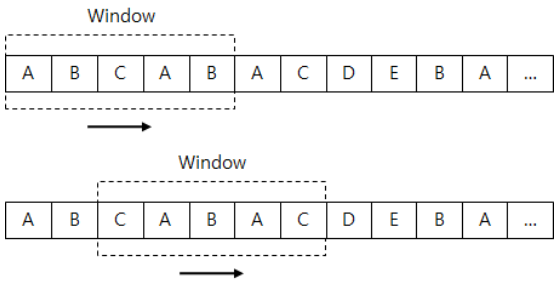
터로부터 발생할 수 있는 이벤트 타입이 유한의 집합이라는 전제를 바탕으로 연구되었다. 이를 이용하여 과거의 자료를 버리지 않고 모두 저장하는 기법을 적용하였고 이를 이용해 과거의 빈발한 아이템들에 대한 정보를 모두 저장할 수 있다. 하지만 이 전제는 한 개의 아이템에 대해 탐사할 때는 유효하지만 빈발한 아이템셋, 즉 패턴을 탐사할 때에는 적용할 수 없다. 왜냐하면 빈발한 패턴은 입력되는 값의 범위가 일정해도 무한한 경우의 수로 생성될 수 있기 때문이다. 물론, 트랜잭션의 길이를 일정한 크기 이상 커지지 않게 하는 방법이 사용될 수 있지만 이는 모든 USN 환경에 적용될 수 없으므로 적합하지 않다.

3. 슬라이딩 윈도우을 적용한 HAPT

3.1 슬라이딩 윈도우

슬라이딩 윈도우란 처음 입력된 자료가 먼저 출력되어 처리되는 FIFO(First In First Out) 처리 방식을 사용한 기법이다. 이 기법은 큐(Queue) 구조를 가지고 있으며 큐 구조에서 데이터가 더 이상 입력될 공간이 없는 오버플로우(Overflow) 현상을 해결하기 위해 먼저 들어온 데이터를 처리하고 이 후에 입력되는 데이터를 입력받아 저장하는 윈도우 기법이다[8].

실시간 데이터 처리 환경에서는 입력되는 자료의 형태가 무한하고 지속적으로 생산되기 때문에 과거의 데이터 값을 계속 유지시킬 수 없다. 이를 해결하기 위해서 이 논문에서는 슬라이딩 윈도우를 사용하여 기준 시간이 지난 자료를 처리하는 방법을 제안한다.



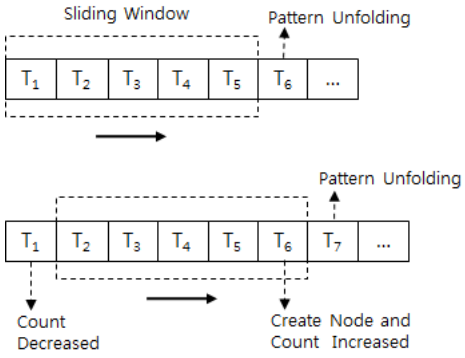
(그림 3) 슬라이딩 윈도우의 예

(그림 3) 은 일정한 윈도우 크기를 사용하여 오른쪽으로 이동하면서 슬라이딩 기법을 사용하는 과정이다. 이 과정에서 윈도우의 이동에 의해 나오게 된 아이템 A와 B는 알고리즘 적용 환경에 따라 삭제, 저장 등의 후처리과정을 거치게 된다.

3.2 HAPT 슬라이딩(HAPT-S)

일정 시간이 지난 과거 데이터를 처리해야 하는 이유는 다음과 같다. 첫째, 무한한 자료가 생성되는 USN 환경에서는 생성되는 데이터의 범위를 알 수 있다는 감독된 시스템(Supervised System)이라는 특징이 있지만, 한 트랜잭션의 길이가 길고 저장되는 아이템의 개수가 많은 환경에서는 부분 순차에 대한 개수가 무수히 늘어나므로 이러한 자료를 모두 저장하는 것은 불가능하다. 둘째, 과거 데이터를 처리하지 않는 것은 시간 속성을 고려하지 않는 방법이기 때문이다. 시간이 지남에 따라 데이터의 가치는 현재 발생하는 데이터와 차별을 두어야 한다. 그렇지 않는다면, 이는 실시간으로 발생하는 데이터를 처리할 때 과거의 데이터와 현재 발생하는 데이터를 같은 성질의 데이터로 간주하고 처리하는, 실시간 스트림 환경의 특징을 고려한 방법이라고 볼 수 없다.

제안하는 기법은 각각의 아이템들에 대한 슬라이딩 기법이 아닌 하나의 트랜잭션을 한 개의 아이템으로 간주하고 이를 시간 순으로 정렬하여 슬라이딩 윈도우를 사용하는 기법이다. 그러므로 슬라이딩 윈도우의 크기가 일정하고 하나의 트랜잭션을 한 개의 아이템으로 간주하여 타겟 이벤트에 대한 발생횟수가 일정하기 때문에 빈발 패턴을 갱신시키는 데에 유리하다. 왜냐하면 슬라이딩 윈도우의 크기가 일정하기 때문에 그에 저장된 전체 트랜잭션의 수도 일정하기 때문이다. 이것은 전체 트랜잭션의 수가 일정할 때 각 아이템의 발생횟수가 변동이 없다면 지지도(Support)는 동일하다는 조건을 만족한다. 이는 전체 트랜잭션의 수를 필요로 하는 빈발항목 탐색과 관련이 있다.



(그림 4) 슬라이딩 윈도우를 적용한 HAPT-S

(그림 4) 는 슬라이딩 윈도우를 적용하여 USN 환경에 적합한 HAPT-S 기법을 나타낸 것이다. 먼저 슬라이딩 윈도우가 이동함에 따라 트랜잭션 T_6이 윈도우에 입력되기 전에 부분순차를 찾아내는 과정(Pattern Unfolding)이 진행되고 이 부분순차들을 HAPT 구조에 새로운 노드를 생성하거나 이미 존재하는 아이템의 발생횟수를 증가시킨다. 그리고 윈도우에서 출력되는 T_1은 각 부분순차들에 대해 발생횟수를 감소시키면서 진행한다. 이러한 과정은 저장되는 데이터의 무분별한 증가를 막아주고 일정한 슬라이딩 윈도우 크기를 유지하면서 빈발 항목을 찾는 과정에서 연산과정을 줄일 수 있도록 해준다.

빈발 패턴 탐색은 슬라이딩 윈도우가 이동할 때, 삭제되는 과거 트랜잭션 데이터와 삽입되는 데이터의 부분순차들의 빈발도만을 계산하여 빈발 패턴 리스트를 갱신해주

면 된다. 그리고 삽입, 삭제되는 데이터 이외에 존재하는 패턴들은 빈발도가 변하지 않게 된다. 왜냐하면 슬라이딩 윈도우에 존재하는 트랜잭션의 수는 일정하고 그에 따라 트랜잭션의 생성 기준인 타겟 이벤트의 빈발 횟수도 일정하기 때문이다.

5. 결론 및 향후연구

이 논문에서는 USN 시스템 환경에서 발생되는 스트림 데이터를 고려한 순차 패턴 탐사 기법으로 슬라이딩 윈도우를 적용한 HAPT-S 기법을 제안하였다. 기존의 HAPT 알고리즘은 과거 데이터를 삭제하지 않고 모두 저장함으로써, 발생 가능한 이벤트와 데이터 영역을 미리 알고 있다는 USN 환경의 특징을 반영하였지만 이는 모든 USN 환경에 적용할 수 없다. 이 논문에서는 과거 데이터에 대한 발생 횟수를 시간 속성을 반영하여 일정한 시간이 지난 후에는 감소하도록 하였고 이는 현재 발생하는 데이터와의 차별성을 부여하며 과거 데이터를 슬라이딩 기법으로 처리하는 방식을 사용하여 시간 속성을 고려한 기법 제안하였다. 향후 연구로는 HAPT-S 기법에 대한 구현을 통하여 USN 환경에 적용 가능한 알고리즘을 연구하도록 한다. 또한 HAPT 및 기존 알고리즘과의 비교실험을 통해 성능을 확인하고 실시간 스트림 데이터를 효율적으로 처리하는데 적합한 시스템을 개발하도록 한다.

참고문헌

[1] M. s. chen. J. Han, and P. S. Yu, "Data Mining: An Overview from a Database Perspective", IEEE Trans. Knowledge and Data Eng., Vol. 9, No. 6, pp.866-883, 1996. 12.

[2] J. Han, and M. Kamber, "Data mining: Concepts and Techniques", Academic Press, 2001.

[3] R. Agrawal and R. Srikant. "Mining sequential patterns", In Proc. 1995 Int. Conf. Data Engineering(ICDE'95), pp.3-14, 1995. 04

[4] J. Pei, J. Han, B. M. Asl, H. Pinto, Q. chen U. Dayal, and M. Hus, "PrefixSpan: mining sequential patterns efficiently by prefix-projected pattern growth", In Proc. Int. Conf. Data engineering(ICDE'01), pp.215-226, 2001.

[5] I. F. Akyiliz, W, Su, Y. Sankarasubramaniam and E. Cayirci, "A survey on sensor networks", IEEE Communications Magagine, pp.102-114, 2002. 08

[6] J. I. Kim, "Real-time Sequential Pattern Mining for USN System", In Proc. 2012 Int. Conf. on Ubiquitous Information Management and Communication (ICUIMC'12), 2012. 2

[7] R. Agrawal and R. Srikant. "Fast algorithms for mining association rules", In Proc. 1994 Int. Conf. Very Large Data Bases(VLDB'94), pp.487-499, 1994. 09

[8] B. J. Wang and Y. Zhan, "A survey and performance evaluation on sliding window for data stream", Communication Software and Networks (ICCSN'11), pp.654-657, 2011. 09