다차원 순차패턴 마이닝을 위한 효율적 알고리즘

이순신* 김은주[0]**, 김명원**
(주)LGCNS*, 숭실대학교 컴퓨터학과**
leess@lacns.com[0], blue7786@ssu.ac.kr, mkim@comp.soongsil.ac.kr

An Efficient Algorithm for Multi-dimensional Sequential Pattern Mining

Sunshine Lee* , Eun Ju Kim[0]**, Myung Won Kim**
LGCNS*, Dept. of Computing, Soongsil University**

요 약

순차패턴 마이닝은 데이터들 속에서 어떤 순차 관계가 들어 있는 패턴을 찾는 것이다. 순차 패턴은 다양한 분야에서 중요하게 쓰인다. 예를 들어, 소비자가 구입한 물품들 간의 순차적인 관계성은 다음에 구입할 물건을 예측하는 데 쓰일 수 있다. 또한 방문 웹 페이지의 순차 패턴은 사용자가 방문하고자 하는 다음 페이지를 예측하는데 중요할 수 있다.

본 논문에서는 다차원 순차패턴을 마이닝하는 새로운 효율적인 알고리즘의 구현에 대해 설명한다. 다차원 순차 패턴 마이닝은 속성-값(attribute-value) 기술을 포함하는 순차 패턴의 연관 규칙을 찾는 것이다. 다음의 두 가지의 현존하는 효율적 알고리즘을 융합하였다: 순차패턴 마이닝을 위한 PrefixSpan 알고리즘과 비 순차패턴 마이닝을 위한 StarCubing 알고리즘. 새로운 알고리즘은 다차원 데이터를 마이닝 하는 StarCubing알고리즘의 효율성을 이용하므로 다차원 순차 데이터를 마이닝 하는데 효율적일 것이다. 실험결과는 제안한 알고리즘이 특히 작은 최소지지도와 작은 cardinality에서 Seq-Dim과 Dim-Seq 같은 현존하는 알고리즘보다 나은 성능임을 보여준다.

1. 서 론

순차 패턴 마이닝(sequential pattern mining)은 데이터들 속에서 어떤 순서 관계가 있는 패턴을 찾는 것으로 전자상거래등 다양한 분야에서 응용되고 있다. 다차원 순차 패턴 마이닝(multi-dimensional sequential pattern mining)이란 속성-값 기술을 포함하는 순차 패턴의 연관 규칙(association rule)을 찾는 방법이다. 대표적인 알고리즘으로는 시간 정보를 고려하지 않는 정보를 이용하는 스타-큐빙(Star-cubing)과 시간 정보를 고려한 정보를 이용하는 프리픽스스팬(PrefixSpan)[2]이 있다. 스타-큐빙은 기존 Apriori 알고리즘이 대량의 데이터에서 수행속도가 느리며, 메모리 소비량이 많은 단점을 보완한다. 프리픽스스팬은 기존 BUC알고리즘에서 최소지지도가 작을때, 수행속도가 급격히 느려지는 문제점을 개선한다.

본 논문에서는 시간 정보를 고려하는 경우와 시간 정보를 고려하지 않은 경우를 동시에 처리할 수 있는 효율적인 다차원 순차패턴 마이닝 기법인 PSStar를 제안한다. PSStar는 효율적인 다차원 순차패턴 마이닝을 위하여 스타-큐빙과 프리픽스스펜 알고리즘을 융합한 방법이다. PSStar는 스타-큐빙과 프리픽스스펜 알고리즘의 장점을 이용하기 때문에 메모리 사용량이 작고, 수행속도가 빠르다.

2. 다차원 순차패턴 마이닝

다차원 순차 마이닝은 다차원 순차 데이터베이스에 저장되어 있는 다차원 정보와 순차 정보를 분석하여 패턴을 찾아낸다. 다차원 정보와 순차 정보를 분석하는 알고리즘은 각각 다르다. 그렇기 때문에 다차원 순차 마이닝을 위해서는 다차원 정보와 순차 정보를 순차적으로 수행하여 다차원 순차 패턴을 찾아야 한다.

다차원 순차패턴 마이닝을 위한 기존의 알고리즘은 UniSeq, Seq-Dim, Dim-Seq[4]이 있다.

UniSeq은 다차원 정보, 순차정보에서 빈번한 패턴을 찾을 때, 하나의 알고리즘으로 패턴을 찾는 것을 말한다. 즉, 다차원 정보를 시간적으로 동시에 구입한 물건으로 가정하고 순차정보와 함께 순차패턴을 찾을 수 있는 프리픽스스팬으로 패턴을 찾는 것을 말한다. 프리픽스스팬이란 연관 규칙을 찾는 데이터 마이닝 기법으로 기존 연관규칙 알고리즘과 달리 후보 패턴을 만들지 않고 프리픽스스팬 트리를 만들어 빈번한 패턴을 찾는 알고리즘이다[2].

Seq-Dim과 Dim-Seq는 다차원 정보와 순차 정보를 둘 중 어떤 정보를 먼저 마이닝을 수행 할 것인가로 구분되어진다. Seq-Dim은 순차 정보를 마이닝 한 뒤 그 순차 정보를 포함하는 다차원 정보만을 이용하여 다차원 마이닝을 수행한다. 반대로 Dim-Seq는 다차원 정보를 먼저 마이닝하고 그 다차원 정보를 가지는 순차 정보들만을 대상으로 마이닝을 수행하는 기법이다.

Seq-Dim과 Dim-seq 모두 BUC알고리즘을 기반으로 하고 있다. BUC 알고리즘은 상향식 큐브 계산을 수행하고 apriori 가지치기(pruning)이 가능하기 때문에 많은 시간상의 이점이 있다. 즉, 계산해 볼 필요가 없는 것들은 미리 계산의 대상에서 제외하여 계산량을 줄이기 때문에 알고리즘 수행시간을 단축한다. 그러나 BUC 알고리즘은 최소 지지도가 작아졌을 때와 cardinality가 작아졌을 때 속도가 많이 느려진다는 단점이 있다. Cardinality는 한 차원의 속성(attribute)이 가질 수 있는 서로 다른(distinct) 값들의 개수를 나타낸다. 예를 들어 과일이라

는 속성에 가지는 값들을 사과, 배, 귤 그리고 딸기가 있으면 그 속성은 cardinality가 4라고 할 수 있다.

BUC 알고리즘은 각 차원의 cardinality만큼의 파티션을 생성해서 그 하위의 영역에 대해서 또다시 파티션을 하게 되는데, 이때 파티션의 수가 적게 되면 그만큼 각 속성값(attribute value)들이 가지는 값들이 최소지지도를 넘을 확률이 더 높아지며 이것은 apriori 가지치기를 할 수 있는 대상이 적어짐을 의미한다. 즉, apriori 가치치기를 수행할 수 없는 것는 cardinality가 작은 데이터의 BUC의 속도가 급속히 느려지게 된다.

3. PSStar

본 논문에서 제안하는 PSStar 알고리즘은 다차원 순차 패턴 마이닝에서 BUC의 속도가 급속히 느려지는 문제를 해결하기 위하여 BUC 대신하여 스타-큐빙 알고리즘을 도입한다. 스타-큐빙은 BUC처럼 Apriori 가지치기를 사용해 시간을 절약한다. 그러므로 최소지지도가 작은 경우에 Apriori 가지치기의 효과를 볼 수 없게 되는 것은 BUC와 동일하다. 그러나 스타-큐빙은 이런 단점을 해결하기 위하여 과거의 계산을 이용하여 계산에 걸리는 시간을 줄이는 Simultaneous Aggregation을 이용한다.

PSStar는 다차원 정보와 순차 정보를 분석해서 다차원 순차패턴을 찾내는 것으로 프리픽스스팬 알고리즘과 스타-큐빙 알고리즘을 합친 이름이다. PSStar 알고리즘은 먼저 Star-Cubing 알고리즘으로 다차원 정보를 마이닝 수행한다. 마이닝을 수행하여 나온 빈발항목으로 순차정보를 Project한다. 하나의 빈발항목에 하나씩 투영된 순차정보 DB를 갖게 된다. 이 각각의 다차원 정보 DB를 가지고 프리픽스스팬을 통해 순차패턴을 찾아낸다. 찾아진 빈발항목과 순차패턴의 집합이 다차원 순차패턴이 된다.

다음은 알고리즘의 수행순서를 나타낸다.

순서1: 빈발항목을 하나 찾는다.

Star-Cubing으로 각각의 큐브를 순회하면서 빈발항목을 찾을 때, 빈발 항목이 하나 찾아지면 다음을 수행한다.

순서2: Projected DB를 만든다.

빈발항목이 하나 찾아질 때 마다 그 빈발항목의 Projected DB를 만든다. Projected DB는 순차정보만을 가지고 있으며, 찾는 방법은 다차원 정보에 빈발항목이 들어 있는 Tuple들을 모두 추출하는 것이다.

순서3: 각 Projected DB를 PrefixSpan으로 순차패턴을 찾아낸다.

만들어진 Projected DB를 가지고 PrefixSpan 알고리즘을 통해 순차패턴을 찾아낸다. Projected DB는 기본 DB보다 훨씬 적은 양이므로 더 빠르게 순차패턴을 찾아낼 수 있다.

순서4: 찾아진 빈발 항목과 순차패턴의 쌍을 만든다.

찾아진 빈발 항목과, 빈발 항목의 Projected DB를 통해 찾아진 순차패턴을 하나의 쌍으로 해서 다차원 순차패턴을 찾아낸다. 각각의 빈발항목에는 그에 맞는 각각의 순차패턴들이 연결되어 있어야 한다.

PSStar는 스타-큐빙을 이용하기 때문에 다른 다차원 순차 패턴 마이닝 알고리즘에 비하여 메모리 사용양이 작고 수행 속도가 빠른 장점을 가진다.

4. 실험결과

4.1 실험 방법

본 논문에서 제안하는 PSStar 알고리즘을 증명하기 위하여 다차원 순차패턴을 찾는 알고리즘 중 가장 빠른 Seq-Dim 알고리즘과 비교하였다. 실험데이터는 순차정보는 IBM에서 사용하는 IBM Dataset Generator를 사용하여 생성하였으며, 다차원 정보는 UIUC에서 사용하는 Dataset Generator를 사용하였다. 성능의 비교는 각각의 최소지지도와 속성의 cardinality에 대한 수행시간(초)를 사용하였다. 실험에 사용된 컴퓨터의 사양은 Pentium4 2.6Ghz, 1GB의 메모리, 60G HDD, Windows XP Home이다.

4.2 실험 결과

실험을 비교하는 방법 중 하나인 최소지지도에 따른 실험 결과는 [그림 1]다음과 같다.

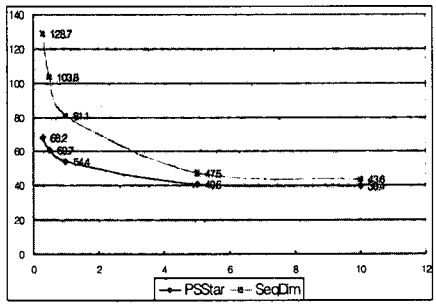

그림 1. 최소지지도에 따른 변화

SeqDim은 최소지지도가 작아질 때 급속히 느려지는 경향을 보인다. 그러나 제안한 PSStar 알고리즘은 서서히 느려지는 모습을 보인다. SeqDim과 PSStar 모두 프리픽스스팬 알고리즘을 사용하지만, 각각의 다른 BUC와 스타-큐빙 알고리즘을 사용하기 때문이다. BUC는 최소지지도가 작은 영역에서 속도가 급속히 느려지는 단점이 있었지만, StarCubing은 BUC에 비해 많이 느려지지 않는다.

속성의 Cardinality 실험 결과는 [그림 2]와 같다. [그림 2]에서처럼 속성의 cardinality가 작아질수록 SeqDim은 급속히 느려지는데 반해, PSStar는 천천히 느려지는 모습을 볼 수 있다.

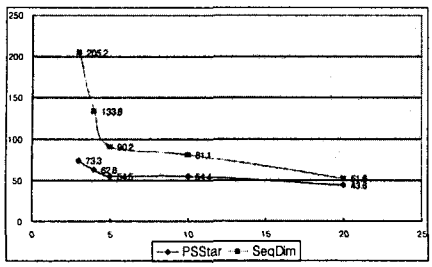

그림 2. cardinality에 따른 변화

Apriori 가지치기만 사용하는 SeqDim의 경우 속성의 cardinality가 작을 경우 계산양이 기하급수적으로 증가한다는 단점이 있다. 그러므로 그림 2에서 처럼 cardinality가 작아질수록 수행속도가 급속도로 증가되게 된다. 그러나 PSStar는 Apriori 가지치기와 Simultaneous 가지치기를 동시에 수행하기 때문에 cardinality가 작아져도 여전히 계산 량을 줄일 수 있다.

PSStar는 SeqDim이 BUC의 단점으로 인해 가질 수 있는 급격한 속도저하를 StarCubing을 수용하므로 해결하였다. StarCubing은 Top-down과 Bottom-up의 Simultaneous Aggregation과 Apriori Pruning을 동시에 수행하므로 BUC의 단점을 해결하는 기초를 제공하며 PSStar의 중요한 두개의 알고리즘 중의 하나로 쓰였다.

그러므로 PSStar 알고리즘은 cardinality가 작은 데이터에서 기존 방법에 비하여 빠른 수행 시간을 보인다. 작은 속성의 cardinality가 가지는 의미는 크다 할 수 있다. 예를 들어, 성별은 남/녀로 cardinality는 2이고, 거주지는 도나 광역시를 기준으로 할 경우 10개에서 20개 정도의 속성값을 가지게 되며, cardinality는 커야 20정도가 된다. 연령대, 최종학력 등 많은 속성들의 cardinality가 작은 값을 가지는 경우가 많다.

4. 결론

본 논문에서는 다차원 정보와 순차 정보를 동시에 마이닝하는 효율적인 융합 방법을 제안한다. PSStar 알고리즘은 먼저 순차정보를 찾기 위해 PrefixSpan으로 순차패턴을 찾아낸다. 그 후 각 순차패턴을 가지고 있는 튜플(tuple)들만을 대상으로 해서 다차원 정보를 스타-큐빙으로 마이닝을 수행한다. 각 빈발항목에 여러 개의 순차패턴이 찾아지므로 빈발항목-순차패턴들 쌍으로 만든다. 모든 순차패턴에 대해 각각의 빈번한 패턴을 찾고 나면, 다차원 순차패턴을 모두 다 찾은 것이다.

PSStar 알고리즘은 Apriori 알고리즘의 단점인 대량의 데이터에서 메모리 소모가 많고 수행 속도가 느린 문제를 개선하는 프리픽스스팬 알고리즘을 사용한다. 또한 BUC 알고리즘의 단점인 최소지지도가 작을 때, 속성의 Cardinality가 작을 때 수행속도가 급격히 느려짐을 개선하는 스타-큐빙 알고리즘을 사용한다.

실험 결과 기존 다차원 순차 패턴 기법인 SeqDim에 비하여 메모리 사용량이 적고 수행속도가 빠르다. SeqDim 알고리즘은 최소지지도가 작아질 때 급격히 수행속도가 늦어지는 결과를 보이지만, 제안한 PSStar 알고리즘은 그 느려지는 비율이 월등히 작으며, 속성의 cardinality가 작아질 때 SeqDim은 급격히 느려지나 PSStar는 그 속도가 천천히 느려짐을 볼 수 있다.

6. 참고 문헌

[1] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In Proc. 1994 Int. Conf. Very Large Data Bases (VLDB'94), pages 487-499, Santiago, Chile, Sep. 1994.

[2] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal and M-C. Hsu. PrefixSpan: Mining Sequential Patterns Efficiently by Prefix-Projected Pattern Growth. In. Proc. 2001 Int. Conf. Data Engineering (ICDE'01), pages 215-224, Heidelberg, Germany, April 2001.

[3] R. Srikant and R. Agrawal. Mining sequential patterns: Generalizations and performance improvements. In Proc. 5th Int. Conf. Extending Database Technology (EDBT'96), pages 3-17, Avignon, France, March 1996.

[4] H. Pinto, J. Han, J. Pei, K. Wang, Q. Chen, and U. Dayal, Multi-dimensional sequential pattern mining, in Proceedings of the 2001 ACM CIKM International Conference on Information and Knowledge Management, ACM, 2001, pp. 81-88.

[5] K. Beyer and R. Ramakrishnan. Bottom-up computation of sparse and iceberg cubes. In Proc. 1999 ACM-SIGMOD Int. Conf. Management of Data (SIGMOD'99), pages 359-370, Philadelphia, PA 1999.

[6] R. Agrawal and R. Srikant. Mining sequential patterns. In Proc. 1995 Int. Conf. Data Engineering (ICDE'95), pages 3-14, Taipei, Taiwan, Mar. 1995.

[7] J. Han, J. Pei and Y. Yin. Mining Frequent Patterns without Candidate Generation. In Proc. 2000 ACM-SIGMOD Int. Conf. Management of Data (SIGMOD'00), pages 1-12, Dallas, TX, May 2000.

[8] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal and M-C. Hsu. PrefixSpan: Mining Sequential Patterns Efficiently by Prefix-Projected Pattern Growth. In. Proc. 2001 Int. Conf. Data Engineering (ICDE'01), pages 215-224, Heidelberg, Germany, April 2001.

[9] M. J. Zaki. SPADE: An Efficient Algorithm for Mining Frequent Sequences. In Proc. of Machine Learning Journal, special issue on Unsupervised Learning (Doug Fisher, ed.), Vol. 42 Nos. 1/2, pages 31-60, Jan/Feb 2001.