ISSN 1598-0170 (Print)
ISSN 2287-1136 (Online)
http://www.jksii.or.kr

다중 최소 임계치 기반 빈발 패턴 마이닝의 성능분석☆

Performance Analysis of Frequent Pattern Mining with Multiple Minimum Supports

양 흥 모[1] 윤 은 일[1*]
Heungmo Ryang Unil Yun

요 약

거대한 데이터베이스로부터 중요하고 의미 있는 정보를 찾아내기 위해 데이터 마이닝 기법들이 사용되며, 패턴 마이닝은 이러한 데이터 마이닝을 위한 중요한 기법 중에 하나이다. 패턴 마이닝은 거대 데이터베이스로부터 유용한 패턴을 찾아내는 기법이며, 패턴 마이닝 분야 중에 하나인 빈발 패턴 마이닝은 데이터베이스에서 최소 임계치 이상의 빈도수를 가지는 빈발 패턴을 마이닝 한다. 전통적인 빈발 패턴 마이닝은 전체 데이터베이스에 대한 단일 최소 임계치를 기반으로 중요 빈발 패턴을 마이닝 한다. 단일 최소 임계치 모델은 데이터베이스 내 모든 아이템이 동일한 특성을 가진다고 암묵적으로 가정한다. 그러나 실제 응용에서는 각 아이템들이 개별적인 특성을 가지고 있을 수 있으며, 따라서 이를 반영한 패턴 마이닝 기법이 요구된다. 데이터베이스 내 아이템들의 이러한 특성이 반영되지 않은 빈발 패턴 마이닝 모델에서, 중요한 희귀 아이템이 포함된 패턴을 마이닝 하기 위해서는 낮은 최소 임계치를 설정해야 한다. 그러나 너무 낮은 최소 임계치는 의미 없는 아이템들을 포함하는 수많은 패턴을 야기한다. 반대로 높은 최소 임계치는 희귀 아이템이 포함된 패턴을 마이닝 하지 못하는 희귀 아이템 문제라 불리는 딜레마가 발생한다. 이러한 문제의 해결을 위한 초기 연구들은 아이템 빈도수에 따라 데이터를 몇 개의 블록으로 분할하거나 관련 희귀 아이템들을 하나의 그룹으로 만드는 방법을 사용한 근사적 접근법을 제안하였다. 그러나 이러한 기법들은 근사적 방법의 적용에 의해 모든 희귀 패턴을 포함한 빈발 패턴을 마이닝 하지 못한다. 다중 최소 임계치를 고려한 패턴 마이닝 모델은 아이템들의 개별적인 특성을 반영하여 희귀 아이템 문제를 해결하기 위해 제안되었다. 다중 최소 임계치 기반의 빈발 패턴 마이닝 모델에서 각 아이템은 MIS (Minimum Item Support)라고 불리는 개별 최소 임계치를 가지며, 아이템들의 데이터베이스 내 빈도수를 기반으로 계산된다. 다중 최소 임계치 모델은 MIS를 통해 수많은 의미 없는 패턴을 생성하지 않고도 손실 없이 모든 희귀 빈발 패턴을 찾아낸다. 한편, 빈발 패턴을 마이닝 하는 과정에서 후보 패턴들이 생성되며, 단일 최소 임계치 모델에서는 각 후보 패턴의 빈도수가 유일한 최소 임계치와 비교된다. 따라서, 희귀 아이템 문제가 발생할 뿐만 아니라 후보 패턴을 구성하는 아이템들의 특성이 고려되지 않는다. 다중 최소 임계치 모델에서는 이 문제를 다루기 위해 후보 패턴을 구성하는 아이템들의 MIS 값 중에서 가장 작은 MIS 값을 해당 후보 패턴의 최소 임계치로 설정하여 패턴 내 아이템들의 특성을 반영한다. 이를 적용하여 효율적으로 희귀 빈발 패턴을 마이닝 하기 위해 트리 구조 기반의 알고리즘은 빈도수 내림차순으로 트리 내 아이템들을 정렬하는 단일 최소 임계치 모델과는 달리 MIS 내림차순으로 아이템들을 정렬하여 마이닝을 수행한다. 본 논문에서는 다중 최소 임계치 기반의 빈발 패턴 마이닝 알고리즘에 대한 특성을 살펴보고, 일반 단일 임계치 기반 알고리즘과의 성능평가를 수행한다. 성능평가는 실행 속도, 메모리 사용량, 그리고 확장성의 관점에서 수행된다. 성능평가 결과, 다중 최소 임계치 기반의 빈발 패턴 마이닝 알고리즘은 희귀 빈발 패턴을 포함한 모든 빈발 패턴을 단일 임계치 기반의 빈발 패턴 마이닝 알고리즘보다 더 빠른 속도로 마이닝 하였으며, 각 아이템의 최소 임계치 정보를 위한 추가적인 메모리를 필요로 하였다. 또한, 비교 알고리즘들은 좋은 확장성 결과를 보였다.

☞ 주제어 : 다중 최소 임계치, 빈발 패턴 마이닝, 희귀 빈발 패턴, 성능평가, 확장성

1 Dept. of Computer Engineering, Sejong University, Seoul, 143-747, Korea
* Corresponding author (yunei@sejong.ac.kr)
[Received 5 August 2013, Reviewed 9 August 2013, Accepted 10 October 2013]
☆ 본 논문은 2013년도 정부 교육과학기술부의 재원으로 한국연구재단(NRF)의 지원을 받아 수행된 연구사업임(No.2013005682 and 20080062611).
☆ 본 논문은 2013년도 인터넷정보학회 춘계학술발표대회 우수논문 추천에 따라 확장 및 수정된 논문임.

ABSTRACT

Data mining techniques are used to find important and meaningful information from huge databases, and pattern mining is one of the significant data mining techniques. Pattern mining is a method of discovering useful patterns from the huge databases. Frequent pattern mining which is one of the pattern mining extracts patterns having higher frequencies than a minimum support threshold from databases, and the patterns are called frequent patterns. Traditional frequent pattern mining is based on a single minimum support threshold for the whole database to perform mining frequent patterns. This single support model implicitly supposes that all of the items in the database have the same nature. In real world applications, however, each item in databases can have relative characteristics, and thus an appropriate pattern mining technique which reflects the characteristics is required. In the framework of frequent pattern mining, where the natures of items are not considered, it needs to set the single minimum support threshold to a too low value for mining patterns containing rare items. It leads to too many patterns including meaningless items though. In contrast, we cannot mine any pattern if a too high threshold is used. This dilemma is called the rare item problem. To solve this problem, the initial researches proposed approximate approaches which split data into several groups according to item frequencies or group related rare items. However, these methods cannot find all of the frequent patterns including rare frequent patterns due to being based on approximate techniques. Hence, pattern mining model with multiple minimum supports is proposed in order to solve the rare item problem. In the model, each item has a corresponding minimum support threshold, called MIS (Minimum Item Support), and it is calculated based on item frequencies in databases. The multiple minimum supports model finds all of the rare frequent patterns without generating meaningless patterns and losing significant patterns by applying the MIS. Meanwhile, candidate patterns are extracted during a process of mining frequent patterns, and the only single minimum support is compared with frequencies of the candidate patterns in the single minimum support model. Therefore, the characteristics of items consist of the candidate patterns are not reflected. In addition, the rare item problem occurs in the model. In order to address this issue in the multiple minimum supports model, the minimum MIS value among all of the values of items in a candidate pattern is used as a minimum support threshold with respect to the candidate pattern for considering its characteristics. For efficiently mining frequent patterns including rare frequent patterns by adopting the above concept, tree based algorithms of the multiple minimum supports model sort items in a tree according to MIS descending order in contrast to those of the single minimum support model, where the items are ordered in frequency descending order. In this paper, we study the characteristics of the frequent pattern mining based on multiple minimum supports and conduct performance evaluation with a general frequent pattern mining algorithm in terms of runtime, memory usage, and scalability. Experimental results show that the multiple minimum supports based algorithm outperforms the single minimum support based one and demands more memory usage for MIS information. Moreover, the compared algorithms have a good scalability in the results.

☞ keyword : Multiple minimum supports, Frequent pattern mining, Rare frequent patterns, Performance evaluation, Scalability

1. 서 론

데이터 마이닝은 거대 데이터베이스 내의 숨겨진 유용한 정보를 의사 결정을 위해 분석 및 발견하는 기술로서, 데이터의 폭발적인 증가와 함께 활발하게 연구되고 있다. 패턴 마이닝은 데이터 마이닝을 위한 중요한 기법 중에 하나로서 대규모 데이터베이스로부터 중요 패턴을 찾아내며, 다양한 분야에서 응용 [2, 12, 17]되고 있다. 패턴 마이닝 분야 중에 하나인 빈발 패턴 마이닝은 데이터베이스에서 최소 임계치 이상의 빈도수를 가지는 빈발 패턴을 마이닝 한다. 전통적인 빈발 패턴 마이닝 [1, 4, 5]은 전체 데이터베이스에 대한 단일 최소 임계치를 기반으로 주요 빈발 패턴을 마이닝 한다. 즉, 단일 최소 임계치 모델은 암묵적으로 데이터베이스 내 모든 아이템들이 동일한 특성을 가지는 것으로 가정한다. 하지만 실세계 응용에서는 아이템들이 개별적인 특성을 가질 수 있으며, 따라서 이를 고려할 필요가 있다 [13]. 만약 이러한 특성을 고려하지 않은 단일 최소 임계치 모델에서 드물게 발생하는 중요한 아이템, 즉 중요 희귀 아이템이 포함된 패턴을 마이닝 하기 위해서는 낮은 최소 임계치를 설정해야 한다. 그러나 너무 낮은 최소 임계치는 의미 없는 아이템들을 포함하는 수많은 패턴을 야기한다. 반대로 높은 최소 임계치는 희귀 아이템이 포함된 패턴을 마이닝 하지 못한다. 이러한 딜레마는 희귀 아이템 문제 [15]라 한다. 예를 들어, 의료 데이터베이스에서 독감은 SARS (Severe Acute Respiratory Syndrome)보다 더욱 빈발하게 발생하며, 공통적으로 발열과 지속적인 기침이라는 증상을 야기한다. 만약 최소 임계치가 높으면 {독감, 발열, 기침}을 발견할 수 있지만, {SARS, 발열, 기침}은 발견할 수 없다. 이러한 패턴을 찾기 위한 낮은 최소 임계치 사용은 수많은 의미 없는 패턴이 마이닝 되는 결과를 초래한다. 이 문제를 해결하기 위한 초기 연구들 [2, 11]은 아이템들의 빈도수에 따라 데이터를 몇 개의 블록으로 분할하거나 관련 희귀 아이템들을 하나의 그룹으로 만드는 방법을 사용했다. 그러나 이는 근사적 방법이며, 손실 없이 모든 패

턴을 마이닝 하지 못한다. 이러한 문제를 다루기 위해, 다중 최소 임계치 기반의 패턴 마이닝 모델이 제안 [6, 9, 13]되었다. 이 모델에서 각 아이템은 MIS (Minimum Item Support)라 불리는 개별 최소 임계치를 가지며, 이를 기반으로 의미 없는 패턴의 생성 및 중요 패턴의 손실 없이 모든 희귀 빈발 패턴을 포함한 빈발 패턴을 마이닝 한다. 본 논문은 2장에서 단일 최소 임계치 기반의 빈발 패턴 마이닝에 대하여 논하고, 다중 최소 임계치 기반의 빈발 패턴 마이닝에 대하여 이야기한다. 3장에서는 단일 및 다중 최소 임계치 기반의 알고리즘의 성능을 분석하고, 4장에서 다중 최소 임계치 기반의 빈발 패턴 마이닝에 대한 결론을 맺는다.

2. 관련 연구

2.1 단일 최소 임계치 기반의 빈발 패턴 마이닝

$I = \{i_1, i_2, ..., i_m\}$를 유한 아이템 집합 그리고 $D = \{T_1, T_2, ..., T_n\}$를 n개의 트랜잭션으로 구성된 데이터베이스라고 하면, I 내 아이템으로 구성된 각 트랜잭션 $T_i \in D$ ($1 \le i \le n$)는 고유한 *TID*로 식별된다. (표 1)은 이러한 트랜잭션 데이터베이스의 예이다.

(표 1) 트랜잭션 데이터베이스 예
(Table 1) An Example of Transaction Database

TID	Transaction
100	*A, B, C*
200	*C, D, E*
300	*A, B, D*
400	*B, C*

또한, 패턴 $P = \{i_1, i_2, ..., i_k\}$ ($1 \le k \le m$)는 k개의 고유한 아이템들로 구성되며, 패턴의 빈도수는 트랜잭션 데이터베이스 내 해당 패턴을 포함하는 트랜잭션의 수를 의미한다. 예를 들어, (표 1)의 트랜잭션 데이터베이스에서 패턴 {*A*, *B*}는 트랜잭션 100과 300에 포함되어 있으므로 빈도수는 2이다. 만약 최소 빈도수 임계치가 주어지면, 단일 임계치 기반의 빈발 패턴 마이닝에서는 데이터베이스로부터 해당 임계치 이상의 빈도수를 가지는 모든 패턴을 찾아낸다. 즉, 하나의 최소 임계치를 기준으로 모든 패턴의 빈도수와 비교하여 모든 빈발 패턴을 마이닝 한다. 안티 모노톤 속성 (anti-monotone property) [1]은 빈발 패턴 마이닝에서 빈발하지 않은 아이템들을 제거하여 효율적으로 마이닝 하기 위해 사용된다. 이 속성은 빈발하지 않은 패턴이 있다면, 해당 패턴의 모든 슈퍼 패턴 (super pattern)들 또한 빈발하지 않다는 것을 의미한다. 예를 들어, 최소 빈도수 임계치가 2로 주어지면, (표 1)의 트랜잭션 데이터베이스에서 패턴 {*E*}는 빈도수가 1이므로 빈발하지 않은 패턴이며, 따라서 {*E, C*}, {*E, D*}, {*E, C, D*} 등의 {*E*}의 슈퍼 패턴들 또한 빈발하지 않으므로 해당 슈퍼 패턴들에 대한 마이닝 작업을 수행하지 않는다.

Apriori [1]는 빈발 패턴 마이닝을 위한 초기 알고리즘이며, 수많은 후보 패턴의 생성 및 다중 데이터베이스 스캔이라는 단점을 지닌다. 이러한 단점을 해결하기 위해, 분할 정복 기법을 기반으로 하는 FP-Growth [5] 알고리즘이 제안되었다. FP-Growth는 데이터베이스를 총 두 번 스캔 하여 FP-Tree (Frequent Pattern Tree)라 불리는 트리 자료구조를 구축하여 모든 빈발 패턴을 마이닝 하며, 효율적인 메모리 사용을 위해 트리 내 아이템들을 빈도수 내림차순으로 정렬한다. 즉, 빈도수가 높은 아이템들을 트리 상위로 배치함으로써 트리 내 노드 개수를 줄여 효율적으로 공간을 사용한다. 이러한 전통적인 빈발 패턴 마이닝 알고리즘들은 단일 최소 임계치를 사용하여 모든 빈발 패턴을 마이닝 한다. 즉, 암묵적으로 데이터베이스 내 모든 아이템들이 동일한 특성을 가진다고 가정한다.

2.2 다중 최소 임계치 기반의 빈발 패턴 마이닝

다중 최소 임계치 기반의 빈발 패턴 마이닝은 데이터베이스 내 개별 아이템에 대하여 MIS (Minimum Item Support)라 불리는 최소 아이템 임계치를 설정하여 사용한다. 본 모델에서, 각 아이템 i의 최소 아이템 임계치 $MIS(i)$는 사용자에 의해 설정되거나 다음의 수식을 통해 결정된다.

$$MIS(i) = \arg\max[\beta \times f(i), LS] \quad (1)$$

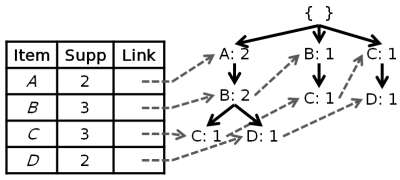
위 수식에서, $f(i)$는 데이터베이스 내 아이템 i의 빈도수 그리고 LS는 최소 아이템 임계치이다. 즉, 빈도수를 기반으로 계산된 최소 아이템 임계치 값이 LS보다 작으면 해당 아이템은 LS 값이 최소 아이템 임계치 값으로 설정된다. 단일 최소 임계치 기반 모델에서 마이닝 과정 중 생성된 후보 패턴의 빈도수를 단일 최소 임계치와 비교하여 빈발 패턴을 마이닝 하는 것과는 달리, 다중 최소 임계치 기반 모델에서는 후보 패턴 내 아이템들의 MIS 값들 중에서 가장 작은 MIS 값을 그 후보 패턴에 대한

최소 임계치로써 사용한다. 예를 들어, 세 개의 아이템 {*fever*, *cough*, *nausea*}로 구성된 데이터베이스에 대해 *MIS*(*fever*) = 10, *MIS*(*cough*) = 5, *MIS*(*nausea*) = 2일 때, 만약 패턴 {*fever*, *cough*}의 빈도수가 4이면, 해당 패턴 내 최소 MIS 값, 즉 *MIS*(*cough*)보다 빈도수가 작으므로 {*fever*, *cough*}는 빈발하지 않다. 반대로 패턴 {*fever*, *cough*, *nausea*}의 빈도수가 3이면, 가장 작은 *MIS*(*nausea*)에 의해 빈발한 패턴이다. 다중 최소 임계치 기반의 빈발 패턴 마이닝은 위의 예와 같이 전통적인 빈발 패턴 마이닝에서 사용되는 안티 모노톤 속성이 만족되지 않는다. 즉, 패턴 {*fever*, *cough*}가 빈발하지 않더라도 패턴을 구성하는 아이템들의 특성에 따라 슈퍼 패턴 {*fever*, *cough*, *nausea*}가 빈발할 수 있다.

다중 최소 임계치를 기반으로 희귀 빈발 패턴을 포함한 빈발 패턴을 마이닝 하기 위해, Apriori 알고리즘 기반의 MSApriori [13]가 제안되었다. MSApriori는 Apriori를 기반으로 하여 수많은 후보 패턴의 생성 및 다중 데이터베이스 스캔의 단점을 지니며, 이로 인해 마이닝 성능이 저하된다. 이러한 문제를 해결하기 위해, FP-Tree 기반의 CFP-Growth [7]가 제안되었다. CFP-Growth 알고리즘은 한 번의 데이터베이스 스캔을 통해 MIS-Tree (Minimum Item Support Tree)라 불리는 트리를 구축한다. 또한, MIS 값들 중에서 가장 작은 값을 MIN이라 정의하고, 이보다 작은 빈도수를 가지는 아이템들에 대한 프루닝 (pruning)을 통해 트리를 재구축한 후 마이닝을 수행한다. 비록 CFP-Growth 알고리즘이 FP-Tree를 기반으로 MSApriori보다 더욱 빠른 속도로 마이닝을 수행하지만, 적용된 분할 정복 기법의 반복 횟수가 많아 마이닝 성능이 저하된다. 이 문제를 다루기 위해, 분할 정보 기법의 반복 횟수의 감소를 통해 마이닝 성능을 개선한 CFP-Growth++ [9]가 제안되었다. CFP-Growth++ 알고리즘은 구축된 전역 트리로부터 검색 공간 분할 시, 분할 검색 공간에 대한 아이템의 MIS 값을 기본 최소 빈도수로 하여 분할된 검색 공간 내에서 프루닝을 통해 반복 횟수를 감소시킨다. 한편, MIS-Tree 구축을 위해 원본 데이터베이스에 대한 스캔을 한 번 수행하며, 이때 각 트랜잭션 내 아이템들은 MIS 내림차순으로 정렬되어 트리에 삽입된다. 전역 MIS-Tree가 구축되면 트리에서 빈발하지 않은 아이템들을 프루닝 하여 트리를 재구축한다. (표 2)와 (그림 1)은 각각 (표 1)의 트랜잭션 데이터베이스에 대한 각 아이템의 최소 아이템 임계치 그리고 구축 및 재구축된 MIS-Tree이다. 이때, 아이템 *E*는 (표 1)의 데이터베이스에서 (표 2)의 아이템들의 MIS 값들 중에서 가장 작은 2보다 작은 빈도수 값인 1을 가지므로 프루닝 되며, (그림 1)은 아이템 *E*가 프루닝 되어 구축된 MIS-Tree이다.

(표 2) 각 아이템의 최소 아이템 임계치
(Table 2) Minimum Item Support for each item

Item	*A*	*B*	*C*	*D*	*E*
MIS	4	3	3	2	2



(그림 1) MIS-Tree
(Figure 1) MIS-Tree

3. 다중 최소 임계치 기반 알고리즘의 성능분석

본 논문에서는 다중 최소 임계치 기반의 빈발 패턴 마이닝 알고리즘인 CFP-Growth++ [9]을 단일 최소 임계치 기반의 빈발 패턴 마이닝 알고리즘인 FP-Growth [5]와 성능평가 및 분석한다. 모든 실험은 3.3 GHz Intel 프로세서와 8GB 메모리 환경의 Windows 7 운영체제에서 수행되었다. 또한, 비교 알고리즘들은 C++ 언어로 구현되었다. 본 장에서는 수행 시간과 메모리 사용량 그리고 확장성을 평가한다. (표 3)은 수행 시간과 메모리 사용량의 성능 평가를 위한 실제 데이터 셋의 특성이다. Retail은 Belgian retail store의 물품 판매에 대한 데이터로서 FIMI 저장소 (http://fimi.ua.ac.be)에서 획득하였으며, BMS-Web-View1 데이터 셋은 작은 닷컴 회사에서 수집된 웹 페이지 클릭 스트림에 대한 데이터로서 KDD-Cup 2000 [10]에서 사용되었다.

(표 3) 실제 데이터 셋의 특성
(Table 3) Characteristics of Real Datasets

Dataset	#Items	Average Length	#Transactions
BMS-Web-view	497	2.5	59,602
Retail	16,469	10.3	88,162

(표 4)는 확장성 평가를 위한 가상 데이터 셋으로서 데이터베이스 내 트랜잭션의 수가 증가한다. 해당 데이터 셋들 IBM의 데이터 셋 생성기 [1]를 통해 생성하였으며,

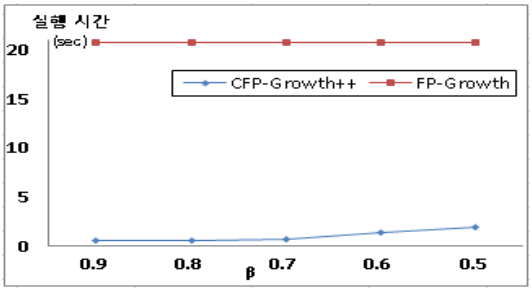
이들 데이터 셋에 대한 성능평가를 통해 트랜잭션 크기의 일정한 증가에 따른 확장성을 평가한다. 모든 데이터 셋에 대한 MIS 값은 수식 (1)을 통해 계산된다.

(표 4) 가상 데이터 셋의 특성
(Table 4) Characteristics of Synthetic Datasets

Dataset	#Items	Average Length	#Transactions
T10I4D200K	10,000	10	200,000
T10I4D400K	10,000	10	400,000
T10I4D600K	10,000	10	600,000
T10I4D800K	10,000	10	800,000
T10I4D1000K	10,000	10	1,000,000

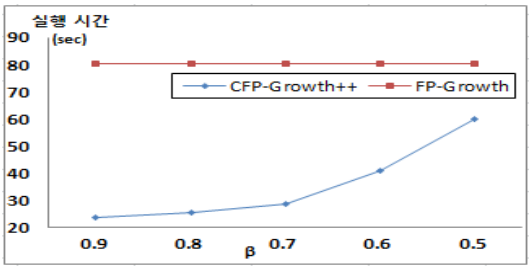
3.1 수행 시간 성능평가

비교 알고리즘들의 수행 시간에 대한 평가는 (표 3)의 실제 데이터 셋을 사용하였으며, 수행 시간은 알고리즘이 시작하는 시점부터 마이닝을 완료하여 최종 패턴이 도출되는 종료 시점까지의 시간을 측정하였다. (그림 2)는 BMS-Webview1 데이터 셋에 대해서 최소 임계치(*LS*)가 33.4일 때 수행 시간의 결과이다. 즉, 빈도수가 최소 33.4 이상인 빈발 패턴들이 마이닝 된다. 단일 최소 임계치(*LS*)를 사용하는 일반 빈발 패턴 마이닝 알고리즘인 FP-Growth는 β에 상관없이 일정한 수행 시간을 필요로 하며, 다중 최소 임계치를 사용하는 마이닝 알고리즘인 CFP-Growth++는 β값이 감소함에 따라 더 많은 수행 시간을 소모한다. (그림 2)의 수행 시간에 대한 결과의 이유는 CFP-Growth++가 데이터베이스 내 빈도수가 높은 아이템을 높은 최소 아이템 임계치를 설정하여 FP-Growth 보다 더 많은 후보 패턴을 프루닝 하기 때문에 더 빠른 속도로 마이닝 과정이 수행된다.



(그림 2) 수행 시간 평가 (BMS-Webview1)
(Figure 2) Runtime test (BMS-Webview1)

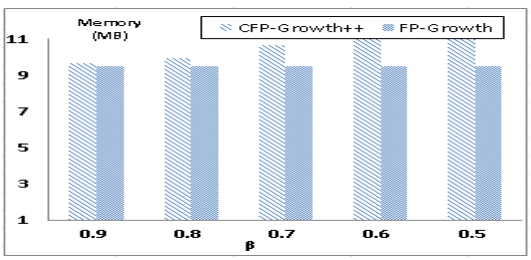
(그림 3)은 Retail 데이터 셋에 대해서 최소 임계치(*LS*)가 3.0일 때 β값을 0.9에서 0.5로 변화시켰을 때의 수행 시간에 대한 결과이다. CFP-Growth++는 β가 감소함에 따라 각 아이템의 MIS 값도 함께 감소하여 수행 시간이 증가하였으며, 반면에 FP-Growth는 일정한 최소 임계치(*LS*)를 사용하여 항상 동일한 수행 시간을 사용하였다.



(그림 3) 수행 시간 평가 (Retail)
(Figure 3) Runtime test (Retail)

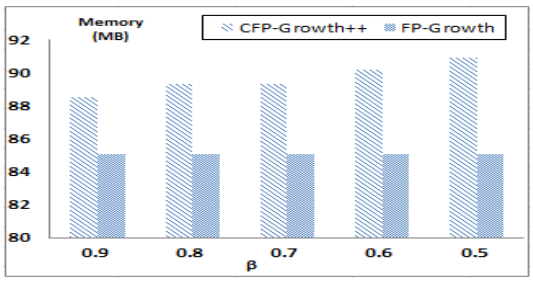
3.2 메모리 사용량 성능평가

다음 평가는 마이닝 수행 시 비교 알고리즘들의 메모리 사용량에 대한 것이다. 성능평가는 (표 3)의 실제 데이터 셋을 사용하였으며, 마이닝이 시작하는 시점부터 종료되는 시점까지 사용된 메모리의 최댓값을 측정하였다. 즉, 알고리즘 수행 시 최대로 사용된 시점의 메모리를 측정한 결과이다. (그림 4)는 BMS-Webview1 데이터 셋에서 최소 임계치(*LS*)가 33.4로 설정됐을 때의 메모리 사용량 결과이다. FP-Growth와는 달리 CFP-Growth++는 트리 구축 시 아이템들을 MIS 내림차순으로 정렬하며, 각 아이템에 대한 개별 최소 임계치 정보를 위한 추가적 메모리가 필요하므로 더욱 많은 메모리 사용량을 필요로 한다. 특히, β값이 감소함에 따라 각 아이템의 MIS 값도 감소하여 검색 공간이 늘어나므로 CFP-Growth++의 메모리 사용량이 증가한다.



(그림 4) 메모리 사용량 평가 (BMS-Webview1)
(Figure 4) Memory usage test (BMS-Webview1)

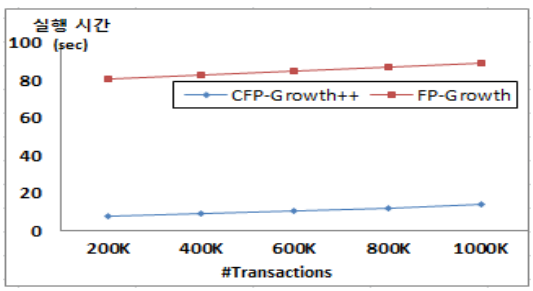
(그림 5)는 Retail 데이터 셋에 대한 최소 임계치(*LS*)가 3.0이고 β 값이 0.9에서 0.5로 변화될 때의 비교 알고리즘들의 메모리 사용량 결과이다. CFP-Growth++는 β 값이 증가함에 따라 더욱 많은 검색 공간을 생성하여 메모리 사용량이 증가한다. 또한, MIS 내림차순으로 정렬된 트리 내 아이템들로 인해 FP-Growth보다 더 많은 노드를 생성하고, 각 아이템에 대한 MIS 정보를 저장하기 위한 저장 공간을 사용함으로써 FP-Growth보다 더 많은 메모리를 사용한다.



(그림 5) 메모리 사용량 평가 (Retail)
(Figure 5) Memory usage test (Retail)

3.3 확장성 평가

본 장에서 마지막으로 확장성에 대한 평가를 수행한다. 확장성 평가는 유사한 정보를 가지는 데이터를 기반으로 트랜잭션 크기가 증가하는 데이터 셋을 사용하여 성능을 평가하는 방법이다. 즉, 점진적으로 커지는 데이터베이스 크기에 따라 알고리즘들이 안정적으로 수행되는지 평가하는 방법이다.



(그림 6) 확장성 평가 (Runtime)
(Figure 6) Scalability test (Runtime)

(그림 6)은 (표 4)의 데이터베이스 크기가 증가하는 가상 데이터 셋을 이용하여 비교 알고리즘들의 수행 시간에 대한 확장성을 평가한 결과이다. 평가에 사용된 데이터 셋 내의 트랜잭션은 20만 개에서 100만 개로 증가되었으며, 최소 임계치는 0.002% 그리고 β는 0.5로 설정하였다. 즉, 20만 개의 트랜잭션이 포함된 T10I4D200K 데이터 셋에 대한 최소 임계치는 4.0으로 설정되었으며, T10I4D400K부터 T10I4D1000K까지 마찬가지로 각각 최소 임계치는 8.0, 12.0, 16.0, 20.0으로 설정되었다. 평가 결과, 두 알고리즘 모두 데이터베이스 크기의 증가에 따라 비슷한 시간 증가율을 보여 주었다. 즉, 좋은 확장성을 보여 주었으며, CFP-Growth++의 수행 시간이 FP-Growth의 수행 시간보다 빠른 결과를 보여 주었다.

4. 결 론

본 논문은 다중 최소 임계치 기반의 빈발 패턴 마이닝 기법에 대하여 분석 및 성능평가를 수행하였다. 성능평가에서는 전통적인 단일 최소 임계치 기반의 빈발 패턴 마이닝 알고리즘인 FP-Growth와 다중 최소 임계치 기반의 빈발 패턴 마이닝 알고리즘인 CFP-Growth++에 대한 마이닝 수행 시간, 메모리 사용량, 확장성 평가를 수행하였다. 성능평가 결과, CFP-Growth++ 알고리즘이 데이터베이스 내 아이템들의 빈도수 특성을 반영한 최소 아이템 임계치를 사용함으로써 FP-Growth보다 더욱 빠른 수행 시간을 보여주었다. 메모리 사용량은 FP-Growth가 트리 내 아이템들을 빈도수 내림차순으로 정렬함으로써 노드 수를 감소시키고, CFP-Growth++에서 각 아이템의 최소 아이템 임계치를 저장하기 위한 추가 공간을 사용함으로써 CFP-Growth++ 알고리즘이 더욱 많은 메모리가 요구되었다. 확장성 평가에서는 CFP-Growth++와 FP-Growth가 증가하는 데이터베이스 크기에 대해 안정적인 수행 시간의 확장성을 보여주었다. 성능평가에 대한 전체적인 분석 결과, CFP-Growth++가 FP-Growth보다 더 빠른 수행 속도와 확장성을 가지고 마이닝 과정을 수행하며, 추가적인 정보를 위한 더 많은 메모리 공간을 필요로 하였다. 실세계 응용에서는 중요하지만 드물게 발생하는 아이템을 포함한 패턴이 중요한 의미를 가지며, 희귀 아이템 문제없이 이를 해결하기 위한 알고리즘이 필요하다. 본 논문에서 소개하고 구현한 CFP-Growth++는 다중 최소 임계치를 기반으로 하여 희귀 아이템 문제없이 FP-Growth보다 더욱 빠른 속도로 희귀 빈발 패턴을 포함한 빈발 패턴을 마이닝 할 수 있다. 이러한 다중 최소 임계치 기반의 빈발 패턴

마이닝이 다양한 분야에 적용된다면 드물게 발생하는 중요한 현상을 분석하는 데 중요한 역할을 할 것으로 예상한다.

참 고 문 헌(Reference)

[1] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules," in Proc. of the 20th Int'l Conf. on Very Large Data Bases (VLDB), pp. 487-499, 1994.

[2] M. Berlingerio, F. Pinelli, and Francesco Calabrese, "ABACUS: frequent pAttern mining-BAsed Community discovery in mUltidimensional networkS," Data Mining and Knowledge Discovery, Vol. 27, No. 3, pp. 294-320, 2013.

[3] J. Han and Y. Fu, "Discovery of Multiple-level Association Rules from Large Databases," in Proc. of the 21th Int'l Conf. on Very Large Database (VLDB), pp. 420-431, 1995.

[4] A.Y.R. González, J.F.M. Trinidad, J.A. Carrasco-Ochoa, and J. Ruiz-Shulcloper, "Mining frequent patterns and association rules using similarities," Expert Systems with Applications, Vol. 40, No. 17, pp. 6823-6836, 2013.

[5] J. Han, J. Pei, and Y. Yin, "Mining Frequent Patterns without Candidate Generation," in Proc. of the 2000 ACM SIGMOD Int'l Conf. on Management of Data, pp. 1-12, 2000.

[6] Y.-H. Hu, F. Wu, and Y.-J. Liao, "An efficient tree-based algorithm for mining sequential patterns with multiple minimum supports," Journal of Systems and Software, Vol. 86, No. 5, pp. 1224-1238, 2013.

[7] Y.-H. Hu and Y.-L. Chen, "Mining Association Rules with Multiple Minimum Supports: a New Mining Algorithm and a Support Tuning Mechanism," Decision Support Systems, Vol. 42, No. 1, pp. 1-24, 2006.

[8] T.C.-K. Huang, "Discovery of fuzzy quantitative sequential patterns with multiple minimum supports and adjustable membership functions," Information Sciences, Vol. 222, pp. 126-146, 2013.

[9] R.U. Kiran and P.K. Reddy, "Novel Techniques to Reduce Search Space in Multiple Minimum Supports-based Frequent Pattern Mining Algorithms," The 14th Int'l Conf. on Extending Database Technology (EDBT), pp. 11-20, 2011.

[10] R. Kohavi, C.E. Brodley, B. Frasca, L. Mason, and Zijian Zheng, "KDD-Cup 2000 Organizers' Report: Peeling the Onion," SIGKDD Explorations (SIGKDD), Vol. 2, No. 2, pp. 86-98, 2000.

[11] W. Lee, S.J. Stolfo, and K.W. Mok, "Mining Audit Data to Build Intrusion Detection Models," in Proc. the 4th Int'l Conf. on Knowledge Discovery and Data Mining (KDD), pp. 66-72, 1998.

[12] Y. Lee and S. Park, "Optimal Moving Pattern Mining using Frequency of Sequence and Weights," Journal of Korean Society for Internet Information, Vol. 10, No. 5, pp. 79-94, 2009.

[13] B. Liu, W. Hsu, and Y. Ma, "Mining Association Rules with Multiple Minimum Supports," in Proc. of the Fifth ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining (KDD), pp. 337-341, 1999.

[14] Y.-H. Liu, "Mining frequent patterns from univariate uncertain data," Data and Knowledge Engineering, Vol. 71, No. 1, pp. 47-68, 2012.

[15] H. Mannila, "Database Methods for Data Mining," in ACM SIGKDD Conf. on Knowledge Discovery and Data Mining (KDD) tutorial, 1998.

[16] G. Pyun and U. Yun, "Performance evaluation of approximate frequent pattern mining based on probabilistic technique," Journal of Korean Society for Internet Information, Vol. 14, No. 1, pp. 63-69, 2013.

[17] M. Shin and W. Paik, "Design and Implementation of Sequential Pattern Miner to Analyze Alert Data Pattern," Journal of Korean Society for Internet Information, Vol. 10, No. 2, pp. 1-13, 2009.

◐ 저 자 소 개 ◑

양 흥 모(Heungmo Ryang)

2011년 충북대학교 컴퓨터공학전공 학사(공학사)
2013년 충북대학교 대학원 컴퓨터과학 석사(공학석사)
2013년~현재 세종대학교 대학원 컴퓨터공학 박사과정(공학박사)
관심분야 : 데이터마이닝, 정보검색, 데이터베이스
E-mail : ryang@sejong.ac.kr

윤 은 일(Unil Yun)

1997년 고려대학교 이학석사(이학석사)
1997년~2006년 한국통신 멀티미디어연구소 전임/선임연구원.
2005년 Texas A&M Univ. 공학박사(공학박사)
2006년~2007년 한국전자통신연구원, 선임연구원
2007년~2012년 충북대학교 전자정보대학 컴퓨터공학부 조교수
2012년~2013년 충북대학교 전자정보대학 소프트웨어학과 부교수
2013년~현재 세종대학교 컴퓨터공학과 부교수
관심분야 : 데이터마이닝, 정보검색, 데이터베이스
E-mail : yunei@sejong.ac.kr