논문 2008-45CI-3-3

트리밍 방식 수정을 통한 연관규칙 마이닝 개선

(Improved Association Rule Mining by Modified Trimming)

황 원 태*, 김 동 승**

(Wontae Hwang and Dongseung Kim)

요 약

본 논문은 2단 샘플링을 통해 정확도는 줄지만 신속하게 연관규칙을 추출하는 새로운 마이닝 알고리즘을 제안한다. 직전 연구인 FAST (Finding Association by Sampling Technique) 기법은 빈발1항목만 최적샘플 형성과정에 적용하여 빈발2항목 및 그이상의 빈발항목을 샘플 추출에 반영하지 못하였다. 이 논문은 그러한 약점을 보완하여 트리밍 과정에서 손실항목과 오류항목의 비중을 동시에 고려하여 다수 빈발항목에 대한 마이닝의 정확성을 높였다. 대표적인 데이터 세트를 써서 실험한 결과 이전연구와 비교해서 동일한 품질하에서 새 알고리즘의 정확도가 향상됨을 확인하였다.

Abstract

This paper presents a new association mining algorithm that uses two phase sampling for shortening the execution time at the cost of precision of the mining result. Previous FAST (Finding Association by Sampling Technique) algorithm has the weakness in that it only considered the frequent 1-itemsets in trimming/growing, thus, it did not have ways of considering mulit-itemsets including 2-itemsets. The new algorithm reflects the multi-itemsets in sampling transactions. It improves the mining results by adjusting the counts of both missing itemsets and false itemsets. Experimentally, on a representative synthetic database, the algorithm produces a sampled subset of results with an increased accuracy in terms of the 2-itemsets while it maintains the same quality of the data set.

Keywords: data mining, association rule mining, random sampling

I. Introduction

Association mining finds association relations among data items in transactions of a large database. Suppose a database D is analyzed that includes N customer transactions $t_1, t_2, \cdots t_N$ of a departmental store. Each transaction t_j consists of a subset of items $I = \{i_1, i_2, \cdots, i_m\}$, and the mining reports some tendency of customers to buy particular items simultaneously when they do shopping. Among the items appearing together in the transactions, only those occurring more than some threshold ratio (called

* 학생회원, 중앙대학교 전자전기공학부(School of Electrical and Electronics Engineering, Chung-Ang University)
** 정회원, 고려대학교 전기전자전파공학부(School of Electrical Engineering, Korea University)
※ 본 논문은 정보통신부 정보통신연구진흥원 지원의 HNRC(HomeNetwork Research Center)-ITRC 사업의 지원으로 수행되었음.
※ 이 논문은 2003년도 및 2005년도 정부재원(교육인적자원부 학술연구조성사업비)으로 한국학술진흥재단의 지원을 받아 수행된 연구임 (KRF-2003-D0049, KRF-2005-041-D00670). 또한 2006.9월 IEEE CIT 학술대회에서 발표한 동일제목의 논문을 확장한 것임
※ This work was supported by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD) (KRF-2005-041-D00670). This paper is expanded from the paper with the same title presented at 2006 IEEE International Conference on Computer and Information Technology, held at Korea University, Sept. 20-22, 2006.
접수일자: 2008년4월26일, 수정완료일: 2008년5월6일

support) to overall transactions called *frequent patterns* are of interest, thus, the mining problem focuses at finding such frequent itemsets. The issues of frequent pattern mining[5, 8] include reducing the database scanning times since the transactional database is usually stored in the disk and scanning the disk data is too costly. In addition, it's attempt to reduce the search space is more important since every subset of I can be frequent and the number of the frequent items is exponential to the size of I.

This research focuses on finding frequent patterns quickly by successive sampling of transaction database. The method adopts a combination of previous FAST (Finding Association by Sampling Technique) algorithm[2] with IFAST[4] (Improved FAST) algorithm to get better association mining in a short time.

II. Previous research

Apriori[1] and its dialect algorithms find the associations in a straightforward manner by successively growing multi-itemsets from 1-itemsets that appear frequently above some rate. However, they require multiple iterations to prune candidate itemsets in the search, thus, the computing time is long. Other algorithms such as FP-growth algorithms [3, 5] have been suggested that avoid multiple scanning of the data base by devising some data structures to remember candidate items in determining multi-items without scanning the data base further.

To find the association in a sampled data base is known a quick and simple way of data mining. However, since it does not go through a through analysis of database in selecting frequent itemsets, some erroneous itemsets are included in the sampled set. Thus, the problems of minimizing both *false itemsets* that should not appear and *missing itemsets* that the algorithm fails to find must be resolved. To remove false data sets and to collect precise itemsets needs more computational effort, resulting in the tradeoff in precision and computing cost.

2.1 FAST algorithm

FAST algorithm[2] selects a limited number of transactions that represent overall database to draw frequent itemsets in a short time. It produces the set of transactions by repeatedly removing *outliers* from each set of k disjoint sets of the database, where k is some integer constant. An outlier transaction is the one resulting in the least discrepancy (with the smallest distance) from S to S_0 when it is removed. In this process, a distance measure $Dist_{L_1}$ can be used in terms of the frequent 1-itemsets defined as

$$Dist_1(S_0,S) = \frac{|L_1(S)-L_1(S_0)|+|L_1(S_0)-L_1(S)|}{|L_1(S_0)|+|L_1(S)|} \quad (1)$$

Here $L_1(S)$ is the set of frequent 1-itemsets of the original database S, whereas $L_1(S_0)$ represents frequent 1-itemsets of the reduced (sampled) set. The measure tells the symmetric difference of the two sets S and S_0. The quality of the reduced set, called *accuracy*, is 1.0 when the two sets are identical, and decreases as they differ, as defined below:

$$accurary = 1 - \frac{|L(S)-L(S_0)|+|L(S_0)-L(S)|}{|L(S)|+|L(S_0)|} \quad (2)$$

In practical implementation, FAST uses either *trimming* or *growing* strategy, where trimming removes outliers from the total samples until the representative set reaches at a given population as given in Table 1[2], however, growing starts from k most matched transactions and selects the next most matched ones in a repeated manner until the total count reaches at a given number.

표 1. 트리밍 알고리즘

Table 1. Trimming algorithm.

```
Obtain a simple random sample S from D;
Compute f(A;S0) for each for each A∈I1(S);
Set S0=S;
while (|S0| > n) {
    Divide S0 into G disjoint groups of h transactions each;
    for each group G {
        Compute f(A;S0) for each A∈I1(S0);
        Set S0=S0 - {t*},   where Dist(S0 - {t*},S)
                                  = min_{t∈G}Dist(S0 - {t},S)
    }
}
```

2.2 Improved FAST

FAST has produced a subset of the database with the accuracy over 0.95, when the accuracy measure accounts for only frequent 1-itemsets[2]. However, if the accuracy measure were modified by including frequent multi-itemsets and applied, FAST algorithm would give a worse result than a random sampling, as shown in Table 2, obtained by a straightforward experiment.

IFAST algorithm[4] is developed to improve the algorithm by taking into account not only frequent 1-itemsets but also 2-itemsets in trimming/growing process. The algorithm uses a new distance measure $Dist_{L_2}$ together with $Dist_{L_1}$ in removing outlier transactions from the database, defined as below:

표 2. 샘플링방식과 FAST 방식에 의해 생성된 빈발 항목 정확도 비교

Table 2. Accuracy of frequent itemsets by random sample and FAST.

	Random sampling	FAST
accuracy of 1-itemsets	0.860	0.995
accuracy of 2-itemsets	0.500	0.461

$$Dist_{L_2}(F,t) = \frac{|L_2(F) - L_2'(t)|}{|L_2(F)|} \tag{3}$$

Here, $L_2(F)$ is the set of frequent 2-itemsets of the database F and $L_2'(t)$ is the set of 2-itemsets of transaction t. To avoid the lengthy procedure of finding 2-itemsets of the original database S, a random sampled subset F is used instead. Thus, those ts with larger $Dist_{L_2}$ due to having a number of 2-itemsets not included in $L_2(F)$ are more likely outlier transactions to be removed in the trimming.

Ⅲ. New algorithm

While IFAST generates a sampled subset that better preserves frequent multi-itemsets as intended, it loses in turn the accuracy due to having poor frequent 1-itemsets. This is because it introduces both missing 1-itemsets and false 1-itemsets. We change the algorithm by modifying IFAST to have *fewer* missing itemsets and allow *some* false itemsets in choosing outliers. The new algorithm employs a modified measure of $Dist_{L_2}$, and selects more transactions that preserve frequent 1-itemsets (with a

표 3. 별종자료를 제거하는 새로운 알고리즘

Table 3. New algorithm to remove outliers.

/* The algorithm finds a *core-set* of *n* transactions from the database *S*.
{Compute the support of each item A $f(A;S)$, $A \in I_1(S)$;
Find frequent two-itemsets $L_2(F)$ by random sampling of *S*.
$S_0 = S$;
while ($|S_0| > n$) {
1. Partition S_0 into X $(=|S_0|/k)$ equal-sized groups;
2. In each group at a time {
2.1. Compute support $f(A;S_0)$, $A \in I_1(S_0)$;
2.2a if ((step % β) != 1)
2.2a-1. Find a transaction having minimum distance $Dist_1(S_0 - t, S)$
/* Let the transaction be t^* */
2.2b else
2.2b-1. Find a transaction having maximum distance $Dist_{L2}'(F, t)$.
/* Let the transaction be t^* */
2.3. Delete the transaction t^* from S_0 (i.e. $S_0 = S_0 - \{t^*\}$);
}
}

certain frequency determined by a parameter β) than IFAST only. Detailed procedure is summarized in Table 3.

3.1 New distance measure L2

Previous distance measure $Dist_{L_2}$ only considers the discrepancy due to missing items in the transaction. However, the discrepancy of two sets becomes also significant if they have false itemsets. Thus, the distance is revised as $Dist_{L_2}'$ by the following formula:

$$Dist_{L_2}'(F,t) = \frac{|L_2'(F) - L_2''(t)| + \alpha\,|L_2''(t) - L_2'(F)|}{|L_2'(F)| + |L_2''(t)|} \quad (4)$$

$L_2'(F)$ is obtained from the frequent 1-itemsets of F which was a randomly sampled set of S, and $L_2''(t)$ is the set of two itemsets included in the transaction t. α reflects the weight of missing 1-itemsets to false 1-itemsets in computing $Dist_{L_2}'$. Our experience tells that false itemsets degrade the accuracy more than the missing ones as shown in Table 4 (that shows a small number of false 1-itemsets induces a number of false multi-itemsets), hence, we choose the value α to be greater than one.

3.2 Combination of FAST and IFAST algorithms

By the efforts to get better 2-itemsets and multi-itemsets in the final subset, IFAST collects samples favorable to frequent 2-itemsets. However, the quality of 1-itemsets in turn degrades severely, resulting in the poor accuracy of the result. Thus, we combine

표 4. T5.I2.D100k 데이터에 대한 무작위 샘플추출방식으로 나타난 손실항목 및 오류항목 현황

Table 4. No. of missing items and false items in a random sampled subset of T5.I2.D100k.

	missing items	false items
1-itemsets	28	14
2-itemsets	7	87
3-itemsets	2	52
4-itemsets	0	15
5-itemsets	0	2

표 5. FAST와 IFAST를 결합하여 트림하는 방식

Table 5. Combination of FAST and IFAST algorithm for trimming.

```
step = 0;
  while (|S0| > n) {
   if (step % β != 1) Run FAST;
     else Run IFAST;
  step++; }
```

FAST and IFAST in the trimming process as a compromise. The relative ratio of applying the two algorithms is controlled by the integer parameter β as shown in Table 5. The following section reports the result.

To compare the different sampled sets, a measure called the *quality* is defined. The index represents the similarity of the core-set (sampled data set) in terms of all frequent single and mulit-itemsets as defined below:

$$quality(S,\ S_0) = \left(\frac{1}{W_c}\right) \times \left(\sum_{n=1}^{k} [\ W_n\{1 - Dist_n(S,\ S_0)\}]\right),$$

$$W_c = \sum_{n=1}^{k} W_n \quad (5)$$

The best value of the quality of 1.0 can be obtained for the perfect (ideal) sampling, and W_n, the weight for the similarity of$n$-itemsets, is chosen as $W_n = \frac{\sqrt{2}}{2^n}$ for simplicity.

IV. Experimental results and discussion

The algorithm was implemented on a computer with AMD Athlon 1.83GHz CPU and 512MB main memory. Input data were synthetically generated by the method used in [1].

The characteristics of the database are as follows: the average number of items in a transaction is 5 ($|T|_{avg}$=5), the average number of potential frequent items is $|I|_{avg}$=2. The minimum support is chosen 0.77% as in [1]. The overall sizes and properties of the database adopted in the experiments are listed in Table 6.

표 6. 실험용으로 생성된 데이터 셋 속성
Table 6. Properties of sets of transactions used in the experiments.

Data	T	I	D	N
T25I20D100kN1	25	20	100,000	1,000
T25I20D200kN1	25	20	200,000	1,000
T25I20D300kN1	25	20	300,000	1,000
T25I20D400kN1	25	20	400,000	1,000

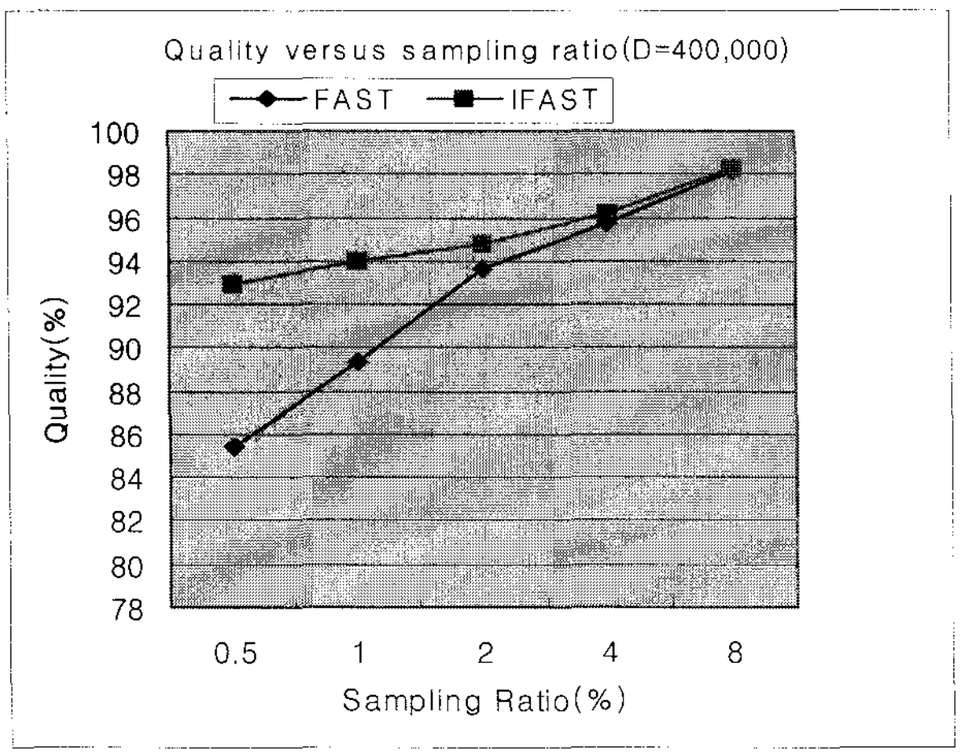

그림 1. 샘플 크기에 따른 품질 비교 (D=400,000)
Fig. 1. Quality of mining result in terms of sampling ratio (D=400,000).

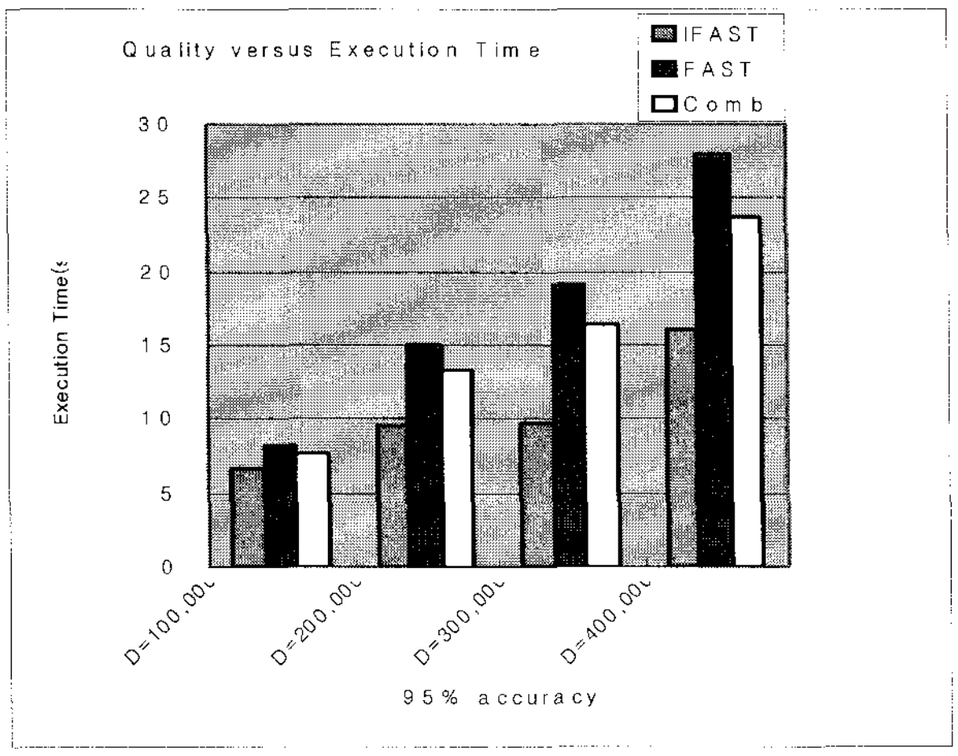

그림 2. 정확도 0.95일때의 알고리즘별 수행시간 비교
Fig. 2. Comparison of execution times with the accuracy of 0.95.

The amount of samples determines the quality of the mining results. However, due to outlier removal process, about less than 5% samples of the original data delivers 95% quality in a reasonable time, as shown in Figure 1 for FAST and IFAST. Figure 2

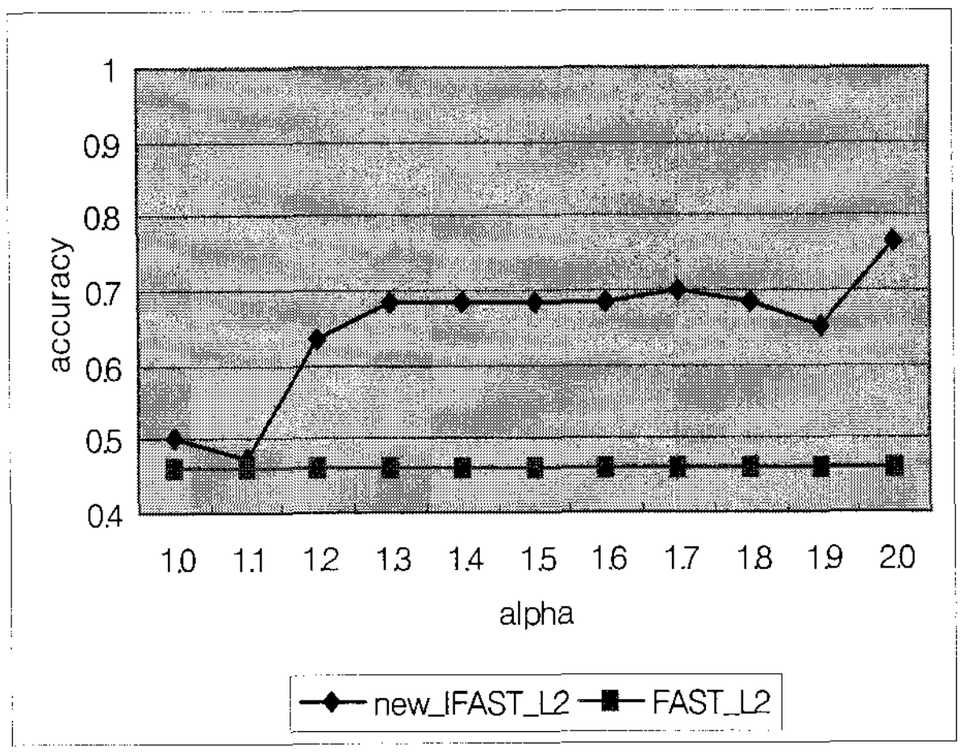

그림 3. α 조정에 따른 2항목 빈발항목 정확도의 변화
Fig. 3. The accuracy of frequent 2-itemsets with respect to α.

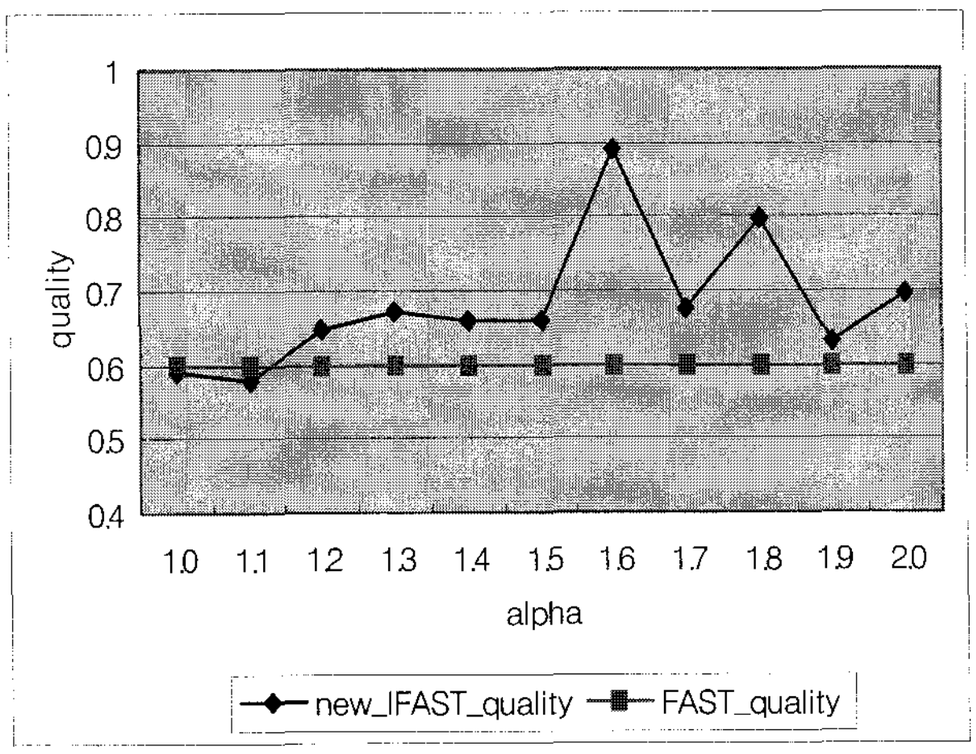

그림 4. α 에 따른데이터 품질지수 비교
Fig. 4. Quality with repect to α.

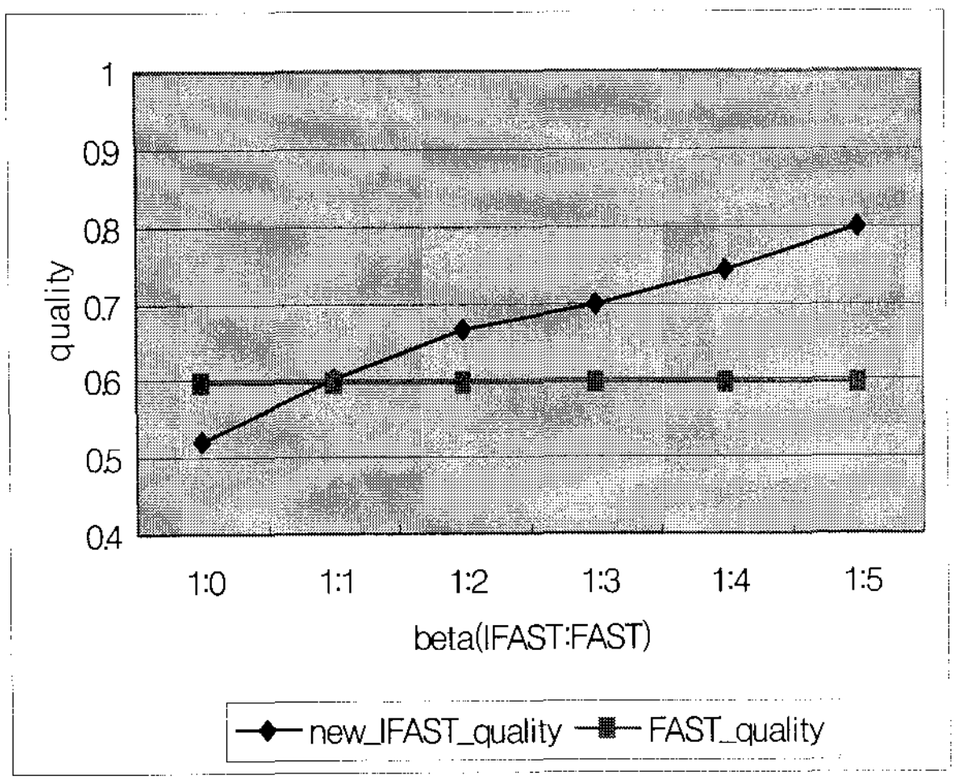

그림 5. β 조정에 따른 품질지수 비교
Fig. 5 Quality of mining with repect to β.

compares the execution times (with the notation "Comb") of new algorithm with various inputs D needed to FAST, IFAST, producing a given quality result. IFAST is fastest, and the new method is in the middle.

In Figure 3 we can observe the improvement of the accuracy with respect to the frequent 2-itemsets in the new algorithm. Various values of α from 1.0 to 2.0 are tested with a fixed value of β=2. The figure shows the maximum improvement of 150% of the algorithm compared to IFAST only.

Figures 4 and 5 plot the quality of the results with various α and β. With (α ,β)=(1.6, 2) the proposed algorithm delivers the best quality output.

From these experimental results, we are convinced that the new algorithm returns better mining results than the previous algorithms.

V. Conclusions

The new mining algorithm by sampling finds a subset of original database that represents the overall database with high accuracy so that the association relationships of the data can be found in a short time. The algorithm enhances the accuracy by determining outliers reflecting both frequent 1-itemsets and 2-itemsets, and maintains the quality by the combination of FAST and IFAST algorithms. Future research will be the parallelization of the algorithm for further speeding up the job, bringing the mining result in a reasonable time even for a large-scale database like a few Terabyte transactions.

References

[1] R. Agrawal and R. Srikant. "Fast algorithms for mining association rules". In *Proc. VLDB Conf.*, 1994, pp.487-499.

[2] B. Chen, P. Haas, and P. Scheuermann, "A new two-phase sampling based algorithm for discovering association rules", *SIGKDD*, 2002.

[3] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation", *SIGMOD*, 2000.

[4] M. Lee and D. Kim, "Modified association rule mining based on two-stage data sampling", *Procs. KISS (Korea Information Systems Society) Conf. on Parallel Processing System*, Vol. 16 No. 1, pp.69-74, Jan. 2005.

[5] G. Liu, H. Liu, Y. Xu, and J.X. Yu, "Ascending frequency ordered prefix-tree: efficient mining of frequent patterns", *Procs. DASFAA 200*

[6] I. Pramudiono and M. Kitsuregawa, "Parallel FP-growth on PC cluster", In *Proc. 7th Pacific Asia Conference on Knowledge Discovery and Data Mining*, pp. 467-473, 2003.

[7] R. Toivonen, "Sampling large databases for association rules", In *Proc. VLDB Conf.*, 1996.

[8] 이문환 (M. Lee), Improved Association Rule Mining Based on FAST(Finding Associations from Sampled Transactions) Algorithm, master thesis, Korea University, July, 2004.

저 자 소 개

김 동 승(정회원)
1978년 서울대학교 전자공학과 학사
1980년 한국과학원 전기및전자공학과 석사
1988년 University of Southern California 공학 박사
1980년~1983년 경북대학교 전임강사
1988년~1989년 University of Southern California Post-doc 연구원
1989년~1995년 포항공과대학 조, 부교수
1995년~현재 고려대학교 부, 정교수
<주관심분야: 병렬 알고리즘, 클러스터 컴퓨팅, 데이터 마이닝>

황 원 태(정회원)
2004년 홍익대학교 전기전자공학부 학사
2006년 고려대학교 전기공학과 석사
2006년~현재 대우일렉트로닉스
<주관심분야: 고성능 컴퓨팅, 데이터마이닝, 컴퓨터 보안>