A Comparative Analysis of Association Rules Mining Algorithms

Komal Khurana[1], Mrs. Simple Sharma[2]

[1]M.tech scholar, [2]Asst.Professor, Department of Computer Science, Manav Rachna International University, Faridabad.
Email id - [1]khuranakomal04@gmail.com, [2]simple.sharma@gmail.com

Abstract- Association rule mining is the one of the most important technique of the data mining. Its aim is to extract interesting correlations, frequent patterns and association among set of items in the transaction database. This paper presents a comparison between different association mining algorithms. All these algorithms are compared according to various factors like type of data set, support counting, rule generation, candidate generation and some other factor .The compared algorithms are presented together with some examples that lead to the final conclusions. Association rules are widely used in various areas such as telecommunication, market and risk management, inventory control etc.

Index Terms- Data Mining, Association rule mining, AIS, Apriori, SETM, AprioriTid, Apriori hybrid.

I. INTRODUCTION

Association rule mining is to find out association rules that satisfy the predefined minimum support and confidence from a given database. The problem is usually decomposed into two sub problems. One is to find those itemsets whose occurrences exceed a predefined threshold in the database; those itemsets are called frequent or large itemsets. The second problem is to generate association rules from those large itemsets with the constraints of minimal confidence. Support and confidence are important measures for association rules.The purpose of Association rule is to find correlation between the different processes,it helps to take decisions and to use the process method effectively.

Generally, an association rules mining algorithm contains the following steps:

i. The set of candidate k-itemsets is generated by 1-extensions of the large (k -1) itemsets generated in the previous iteration.

ii. Supports for the candidate k-itemsets are generated by a pass over the database.

iii. Itemsets that do not have the minimum support are discarded and the remaining itemsets are called large k-itemset.

There are douzines of algorithms used to mine frequent itemsets. Some of them, very well known, started a whole new era in data mining. They made the concept of mining frequent itemsets and association rules possible. Others are variations that bring improvements mainly in terms of processing time. The algorithms vary mainly in how the candidate itemsets are generated and how the supports for the candidate itemsets are counted.

II. ASSOCIATION RULE MINING ALGORITHMS

The problem of discovering association rules was first introduced and an algorithm called AIS was proposed for mining association rules. For last few years many algorithms for rule mining have been proposed. Most of them follow the representative approach of Apriori algorithm. Various researches were done to improve the performance and scalability of Apriori.

III. AIS ALGORITHM

The AIS algorithm was the first algorithm proposed for mining Association rules. AIS algorithm consists of two phases. The first phase constitutes the generation of the frequent itemsets. This is followed by the generation of the confident and frequent association rules in the second phase. The drawback of the AIS algorithm is that it makes multiple pases over the database. Further more,it generate and counts too many candidate itemsets that turn out to be small,which requires more space and waste much efforts that turned out to be useless

k-itemset	An itemset having k items
L_K	Set of large k-itemsets (those with minimum support) Each member of this set has two fields: 1.itemset 2. Support count
C_K	Set of candidate k-itemsets (potentially large itemset) Each member of this set has two fields: 1.itemset 2. Support count
$\bar{C}_K$	Set of candidate k-itemsets when the TIDS of the generating transactions are kept associated with the candidates

Figure: Notation Table

Consider the database in the fig. and assume that the minimum support is 2 database

TID	Items
100	1,3,4
200	2,3,5
300	1,3,2,5
400	2,5

Itemset	Support
{1}	2
{2}	3
{3}	3
{5}	3

C1

Itemset	TID
{1,3}	100
{1,4}	100
{3,4}	100
{2,3}	200
{2,5}	200
{3,5}	200
{1,2}	300
{1,3}	300
{1,5}	300
{2,3}	300
{2,5}	300
{3,5}	300
{2,5}	400

C2

Itemset	Support
{1,3,4}	1
{2,3,5}	2
{1,3,5}	1

Figure 2: Example OF AIS

IV. SETM ALGORITHM

The SETM algorithm was motivated by the desire to use SQL to compute large itemsets. Like AIS, In SETM algorithm candidate itemsets are generated on the fly as the database is scanned but counted at the end of the pass. It thus generates and counts every candidate itemset that the AIS algorithm generates. However, to use the standard SQL join operation for candidate generation, SETM separates candidate generation from counting. It saves a copy of the candidate itemset together with the TID of the generating transaction in a sequential structure. At the end of the pass, the support count of candidate itemsets is determined by sorting and aggregating this sequential structure.

Database

TID	Items
100	1,3,4
200	2,3,5
300	1,3,2,5
400	2,5

C1

Itemset	Support
{1}	2
{2}	3
{3}	3
{5}	3

Itemset	Support
{1,3}	2
{1,4}	1
{3,4}	1
{2,3}	2
{2,5}	3
{3,5}	2
{1,2}	1
{1,5}	1

C2

Itemset	TID
{1,3,4}	100
{2,3,5}	200
{1,3,5}	300
{2,3,5}	300

Figure 3: Example of SETM

V. APRIORI ALGORITHM

This pioneering work appeared first in 1994 and remained the standard reference of all algorithms for finding the association rules. It outperformed the previous algorithm by a great margin. The major difference in Apriori was the much less candidate set of itemsets it generates for testing in every database pass. The search for association rules is guided by two parameters: support and confidence.Apriori returns an association rule if its support and confidence values are above user defined threshold values. The output is ordered by confidence. If several rules have the same confidence then they are ordered by support. Thus apriori favors more confident rules and characterises these rules as more interesting.The apriori Mining process is composed of two major steps. The first one (generating frequent item sets) of the apriori algorithm simply counts item occurrences to determine the large 1-itemsets. This step can be seen as supportbased pruning, because only item sets with at least minimum support were considered. The second step is the generation of rules out of the frequent item sets. In this step confidencebased pruning is applied.Rule discovery is straightforward.
Consider the database in fig. and assume that minimum support count is 2

TID	Items
100	1,3,4
200	2,3,5
300	1,3,2,5
400	2,5

Database

Itemset	Support
{1}	2
{2}	3
{3}	3
{5}	3

C1

Itemset	Support
{1,2}	1
{1,3}	2
{1,5}	1
{2,3}	2
{2,5}	3
{3,5}	2

C2

Itemset	Support
{2,3,5}	2

Figure 4: Example of Apriori

VI. APRIORITID ALGORITHM

The AprioriTid algorithm also uses the apriori-gen function to determine the candidate itemsets before the pass begins. The interesting feature of this algorithm is that the database D is not used for counting support after the first pass. It is not necessary to use the same algorithm in all the passes over the data. Apriori still examines every transaction in the database. On the other hand, rather than scanning the database, AprioriTid scans Ck for obtaining support counts, and the size of Ck has become smaller than the size of the database.Based on these observations AprioriHybrid algorithm has been designed. This uses Apriori in the initial passes and switches to AprioriTid

Consider the minimum support count is 2

TID	Items
100	1,3,4
200	2,3,5
300	1,3,2,5
400	2,5

Datebase

TID	Items
100	1,3,4
200	2,3,5
300	1,3,2,5
400	2,5

C1

Itemset	Support
{1}	2
{2}	3
{3}	3
{5}	3

L1

Itemset	Support
{1,2}	1
{1,3}	2
{1,5}	1
{2,3}	2
{2,5}	3
{3,5}	2

C2

TID	Itemsets
100	{1,3}
200	{2,3},{2,5},{3,5}
300	{!,2},{!,3},{1,5} {2,3},{2,5},{3,5}
400	{2,5}

Itemset	Support
{1,3}	2
{2,3}	2
{2,5}	3
{3,5}	2

Imset	support
{2,3,5}	2

C3

TID	Itemset
200	{2,3,5}
300	{2,3,5}

Itemset	support
{2,3,5}	2

Figure 5: Example of AprioriTid

VII. APRIORIHYBRID ALGORITHM

It is not necessary to use the same algorithm in all the passes over the data.Apriori still examines every transaction in the database. On the other hand, rather than scanning the database, AprioriTid scans Ck for obtaining support counts, and the size of Ck has become smaller than the size of the database.Based on these observations AprioriHybrid algorithm has been designed Figure 7 shows the execution times for Apriori and AprioriTid for different passes. In the earlier passes, Apriori does better than AprioriTid. However, AprioriTid beats Apriori in later passes, the reason for which is as follows. Apriori and AprioriTid use the same candidate generation procedure and therefore count the same itemsets. In the later passes, the number of candidate itemsets reduces However; uses Apriori in the initial passes and switches to AprioriTid in the later passes.

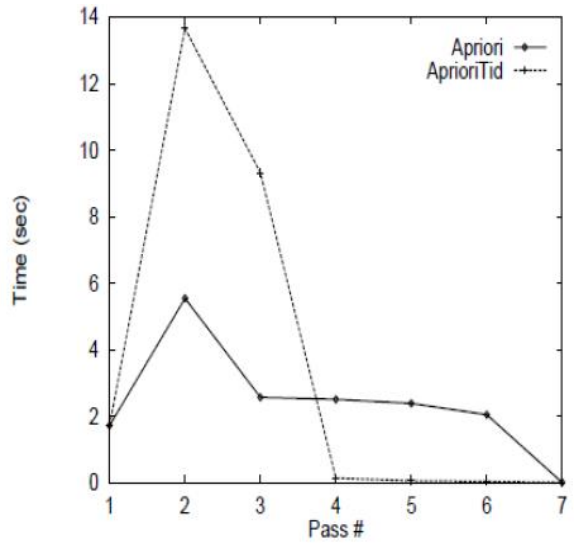

Figure 7: Per pass execution times of Apriori and AprioriTid (min. support=0.75 sec)

VIII. CONCLUSION

This paper represents comparison of five association rule mining algorithms:AIS, SETM, Apriori, AprioriTid and AprioriHybrid. The AprioriTid and AprioriHybrid have been proposed to solve the problem of apriori algorithm. From the comparison we conclude that the AprioriHybrid is better than Apriori and AprioriTid, because it reduced overall speed and improve the accuracy.

Characterstic	AIS	SETM	Apriori	Apriori-Tid	Apriori hybrid
Data support	Less	Less	Limited	Often suppose large	Very large
Speed in initial phase	Slow	Slow	High	Slow	High
Speed in later phase	Slow	Slow	Slow	High	High
Accuracy	Very less	Less	Less	More accurate than apriori	More accurate than apriori-tid

REFERENCES

[1] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. Research Report RJ 9839, IBM Almaden Research Center, San Jose, California, June 1994.

[2] R. Agrawal, T. Imielinski, and A. Swami Database mining: A performance perspective.IEEE Transactions on Knowledge and Data Engineering, 5(6):914{925, December 1993. Special Issue on Learning and Discovery in Knowledge Based Databases.

[3] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In Proc. of the ACM SIGMOD Con- ference on Management of Data, Washington, D.C., May 1993.

[4] Association Rules Mining: A Recent Overview:Sotiris,Kotsiantis,DimitrisKanellopoulos ,Educational Software Development LaboratoryDepartment of Mathematics, University of Patras, Greece.

[5] Margaret H.Dunham,"Data mining Introductory and Advanced Topics",Pearson Education 2008.

[6] J. Han, Y. Cai, and N. Cercone. Knowledge discovery in databases: An attribute oriented approach. In Proc. of the VLDB Conference, pages 547{559, Vancouver, British Columbia, Canada, 1992.

[7] H. Mannila, H. Toivonen, and A. I. Verkamo. Efficient algorithms for discovering association rules. In KDD-94: AAAI Workshop on Knowledge Discovery in Databases, July 1994.

[8] C. Győrödi, R. Győrödi. "Mining Association Rules in Large Databases". *Proc. of Oradea EMES'02*: 45-50, Oradea, Romania, 2002.

[9] J. Han, M. Kamber, "Data Mining Concepts and Techniques", Morgan Kaufmann Publishers, San Francisco, USA, 2001, ISBN 1558604898.