IJCSE International Journal of Computer Sciences and Engineering Open Access
Survey Paper Volume-5, Issue-3 E-ISSN: 2347-2693

# Survey on Association Rule Mining and Its Approaches

M. Shridhar[1*], M. Parmar[2]

[1*]Department of Computer Science and Engineering, Madhav Institute of Technology and Science, Gwalior
[2]Department of Computer Science and Engineering, Madhav Institute of Technology and Science, Gwalior

e-mail: *moksha.itm@gmail.com, Mob.: 8989701151*

**Available online at: www.ijcseonline.org**

Received: 18/Feb/2017 Revised: 27/Feb/2017 Accepted: 19/Mar/2017 Published: 31/Mar/2017

***Abstract*— Apriori calculation has been basic calculation in association rule mining. Principle proposition of this calculation is to discover valuable examples between various arrangements of information. It is the least complex calculation yet having numerous downsides. Numerous specialists have been accomplished for the improvement of this calculation. This paper does a study on couple of good improved methodologies of Apriori calculation. This will be truly exceptionally supportive for the up and coming specialists to locate some new thoughts of this methodology.**

***Keywords- component Apriori algorithm ,frequent pattern, association rule mining. Support, minimum support threshold, multiple scan. FP Growth algorithm,regression technique.***

## I. INTRODUCTION

Association rule mining is a technique for finding engaging relations between factors in vast databases. A case of an affiliation govern would be "If a client purchases twelve eggs, he is 80% liable to likewise secure milk."Association rule mining (ARM) has transform into one of the center information mining errands. Association rules are shaped by investigating information for continuous if/then examples and utilizing the criteria support and certainty to perceive the most vital connections. Support is a recommendation of how habitually the things show up in the database. Certainty demonstrates the numeral of times the if/then articulations have been observed to be valid.

In information mining, affiliation tenets are useful for examining and anticipating client conduct. They have a huge influence in shopping bushel information investigation, item bunching, and index outline and collect design. ARM is an undirected unsupervised information mining strategy which takes a shot at variable length information, and produces evident and justifiable outcomes.

Association rule mining has been comprehensively utilized as a part of various application areas. One of the best perceived is the business field wherever the finding of procurement examples or relationship between items is to a great degree valuable for basic leadership and for powerful promoting. In the earlier years the application territories have expanded altogether.

A few cases of current applications are discovering designs in organic databases, extraction of data from programming building measurements or acquiring client's profiles for net framework personalization. Generally, affiliation examination is viewed as an unsupervised method, so it has been connected in information disclosure errands.

Late reviews have demonstrated that learning discovering calculations, for example, affiliation manage mining, can be viably utilized for forecast in arrangement issues. Most of the exploration endeavors in the extent of the affiliation rules have been arranged to rearrange the administer set and to enhance the calculation execution. In any case, these are not the Just Problem that can be found when guidelines are produced and utilized as a part of distinctive area.

The principle disadvantages of the association rule mining are the accompanying:

- Obtaining non intriguing tenets
- Huge number of found principles
- Low calculation execution

## II. RELATED WORKS

Association rule mining is an information mining errand to distinguishes connections among things inside a value-based database. Association rules have been widely examined in the writing for their part in a few application areas, for example, Market Basket Analysis (MBA), recommender frameworks. Conclusion choices bolster, media transmission,

interruption recognition, and so on. The capable revelation of such principles have been a key concentration amid the information mining research group. The standard apriori calculation has been altered for the change of association rule mining calculations. Association rule digging for Recommender Systems. The creator analyzed the utilization of affiliation lead mining as a crucial strategy for cooperative recommender frameworks. Association rules have been use with sensation in different areas. By and by, most by and by existing association rule mining calculations were planned on account of market wicker container investigation. They portray a community proposal procedure in light of a novel calculation particularly intended to uncover affiliation rules for this method of reasoning. The primary advantage of their proposed approach is that their calculation does not require minimal support to be indicated ahead of time. To a specific degree, an objective range is particular for the quantity of tenets, and the calculation adjusts the base support for all clients with the point of obtaining a control set whose size is into the coveted range. In addition they utilized relationship between clients and in addition relationship between things in making proposals. The test estimation of a framework in view of their calculation found that its execution is broadly superior to anything that of conventional connection based approach.

In [2] proposed a viable information digging approach for finding Adaptive-Support Association Rules (ASAR) from databases. Versatile bolster affiliation principles are obliged affiliation rules with reason to collective proposal frameworks. To discover affiliation rules for suggestion frameworks, a specific estimation of target thing in affiliation guidelines is typically expected and no base support is indicated ahead of time. Contingent upon the size monotonocity of affiliation rules diminishes when the base bolster builds, a compelling calculation utilizing variable stride measure for deciding least support and subsequently versatile bolster affiliation tenets is created.
The essential assignment in some affiliated arrangement approach is mining of the association rule the show. Numerous examinations have uncovered that the base support decide a critical part in developing a flawless classifier. With no data about the things and their recurrence, client offered help measures are inadmissible, not frequently may they match.

In [2] built up a procedure called Dynamic Adaptive Support Apriori (DASApriori) to register the base bolster utilized for acquiring class affiliation governs and to build an uncomplicated and flawless classifier. The association rules describes a huge class of information that can be uncovered from information distribution centers. Show inquire about endeavors are focused on finding efficient methods for deciding these standards from gigantic databases. At an indistinguishable moment from these databases build up, the found tenets are required to be affirmed and it is important to find new principles to the information base. As mining again each time the database create is clumsy, approaches utilized for incremental mining are being contemplated. Their fundamental design is to decrease outputs of the more seasoned database by misusing the go-between information worked amid the past mining exercises.

In [2] utilized expansive and applicant itemsets with their checks in the senior database and inspected the development to find which rules keep up to overcome and which one is not prevailing in the mind boggling database. It is additionally found that new standards for the incremental and refreshed database. The calculation is versatile in nature, as it finish up the way of the addition and dodges by and large if conceivable, different outputs of the incremental database. Another striking component is that it doesn't require different outputs of the senior database [4].

## III. TYPES OF ASSOCIATION RULE

### A. Positive Association Rule Mining

In [3] describes the classical association rules consider only items enumerated in transactions of the dataset. The positive relationship can be found between the set of items. The rules are generated from the positive related items.These rules are referred to as positive association rules. Most of the algorithms were developed for generating positive associations between items. These are useful to decision making. The positive rules are classified as follows:

**1). Boolean association rule: -** It is a rule that checks whether an item is present or absent. There are three types of Boolean association rule:
a. Quantitative
b. Constrained rules
c. Sequential rules

**2). Qualitative association rule: -** It describes associations between quantitative items or attributes. Generally, quantitative values are partitioned into intervals.
Example: Age(X,"30..39") $\wedge$ income(X,"80K..100K") ➔ buys(X, High Resolution TV)

**3). Spatial association rule*:* -** Spatial association rule is a rule indicating certain association relationship among a set of spatial and possibly some non spatial predicates.

**4). Temporal association rule:-** Temporal association rule mining is to discover the valuable relationship among the items in the temporal database.

### B. Negative Association Rule Mining

In [3] describes Negative association rules also consider the same items, but in addition the item also considers which were absent from transactions. The negative rules are generated from infrequent itemsets. These rules play some important role in decision-making. These are useful in market basket analysis to identify products that conflict with each other or products that complement each other. This is a difficult task, due to the fact that there are essential differences between positive and negative rule mining.

### C. Constraint based Association Rule Mining

In [3] describes the constraints were applied during the mining process to generate only those association rules that are interesting to users instead of all the rules. By doing this lots of cost of mining those rules that turned out to be not interesting can be saved. Usually constraints are provided by users. The constraints are classified as follows:

- Knowledge based constraints
- Data constraints

### D. Multilevel Association Rule Mining

Association rule generated from mining data at multiple abstraction levels are called multilevel association rules. It can be mined efficiently under the support confidence framework. A variety of ways include for maintaining min support at each level. Some of them are: -

- Uniform min support for all level
- Reduced min support at each level
- Item or group based min support

## IV. FREQUENT PATTERN MINING

Mining Patterns are set of thing, arrangements, chart or structures that show up in a dataset. The recurrence of example is no less than a client determined limit that is called visit example or thing set. Finding successive examples assumes a basic part in association rule mining, arrangement, grouping, and other information mining tasks. Frequent Pattern mining was beginning proposed by Agarwal for market basket analysis investigation in the sort of association rule mining. The fundamental frequent pattern algorithms are classified into two ways as follows:

- Candidate generation approach (E.g. Apriori algorithm)
- Without candidate generation approach (E.g. FP-growth algorithm)

### A. Candidate Generation Approach

#### 1). Apriori Algorithm

It was proposed by R AGRAWAL AND R SRIKANT in 1994 for mining frequent item sets.The name of this algo is based on the fact that algo uses prior information of frequent item set properties.

This algo works on iterative approach or level wise approach i.e., the frequent item set of size lk+1 can be form using lk. It is used to find the frequent item sets among the given number of transactions. The search proceeds level-by-level as follows:

a) First determine the set of frequent 1-itemset; L1
b) Second determine the set of frequent 2-itemset using L1: L2
c) Etc.

The complexity of computing Li is O (n) where n is the number of transactions in the transaction database. Reduction of search space:

a) In the worst case what is the number of item sets in a level Li?
b) Apriori uses "**Apriori Property**":

**EXAMPLE**:

| TRANSACTION-ID | ITEMS- BOUGHT |
|---|---|
| 1 | A,B,C |
| 2 | A, C |
| 3 | A, D |
| 4 | B,E,F |

| FREQUENT PATTERN | SUPPORT |
|---|---|
| A | 75% |
| B | 50% |
| C | 50% |
| A , C | 50% |

**2). Apriori Property:-**

All non empty subsets of a frequent item set are frequent it means if{A,B,C} is frequent then its subset should be frequent. Apriori algo works on two steps-

**a). Join step:**

- C k is generated by joining Lk-1with itself
  $L_{k-1} \infty L_{k-1}$
- Given $l_1$ and $l_2$ of $L_{k-1}$
  $L_i=l_{i1},l_{i2},l_{i3},\ldots,l_{i(k-2)},l_{i(k-1)}$
  $L_j=l_{j1},l_{j2},l_{j3},\ldots,l_{j(k-2)},l_{j(k-1)}$
  Where $L_i$ and $L_j$ are sorted.
- $L_i$ and $L_j$ are joined if there are different (no duplicate generation). Assume the following:
  $l_{i1}=l_{j1}$, $l_{i2}=l_{j1}$, …, $l_{i(k-2)}=l_{j(k-2)}$ and $l_{i(k-1)} < l_{j(k-1)}$
- The resulting itemset is:
  $l_{i1},l_{i2},l_{i3},\ldots,l_{i(k-1)},l_{j(k-1)}$
- Example of Candidate-generation:
  $L_3$={abc, abd, acd, ace, bcd}

Self-joining: $L_3 \infty L_3$
abcd from abc and abd
acde from acd and ace

**b). Prune Step:**
Any (k-1)-itemset that is not frequent cannot be a subset of a frequent k-itemset

- $C_k$ is a superset of $L_k$ ➔ some itemset in $C_k$ may or may not be frequent.
- $L_k$: Test each generated itemset against the database:
  a) Scan the database to determine the count of each generated itemset and include those that have a count no less than the minimum support count.
  b) This may require intensive computation.

- Use Apriori property to reduce the search space:
- Any (k-1)-itemset that is not frequent cannot be a subset of a frequent k-itemset.
- Remove from Ck any k-itemset that has a (k-1)-subset not in Lk-1 (itemsets that are not frequent)
- Efficiently implemented: maintain a hash table of all frequent itemset.

**Example of Candidate-generation and Pruning:**

$L_3$={abc, abd, acd, ace, bcd}
Self-joining: $L_3 \infty L_3$
abcd from abc and abd
acde from acd and ace

**Pruning:**
acde is removed because ade is not in $L_3$
$C_4$={abcd}

**EXAMPLE:**

Min support=2

| T_id | ITEMSETS |
|---|---|
| 1 | A,C,D |
| 2 | B,C,E |
| 3 | A,B,C,E |
| 4 | B,E |

**Step 1**:-Find C1

| ITEMSET | SUPPORT |
|---|---|
| {A} | 2 |
| {B} | 3 |
| {C} | 3 |
| {D} | 1 |
| {E} | 3 |

**Step 2**:-Find L1

| ITEMSET | SUPPORT |
|---|---|
| {A} | 2 |
| {B} | 3 |
| {C} | 3 |
| {E} | 3 |

**Step 3**:-Find C2

| ITEMSET | SUPPORT |
|---|---|
| A,B | 1 |
| A,C | 2 |
| A,E | 1 |
| B,C | 2 |
| B,E | 3 |
| C,E | 2 |

**Step 4**:- Find L2

| ITEMSET | SUPPORT |
|---|---|
| {A,C} | 2 |
| {B,C} | 2 |
| {B,E} | 3 |
| {C,E} | 2 |

**Step 5**:- Find C3

| ITEMSET | SUPPORT |
|---|---|
| {B,C,E} | 2 |

**Step 6**:- Find L3

| ITEMSET | SUPPORT |
|---|---|
| {B,C,E} | 2 |

**3). Advantages and Disadvantages of Candidate Generation Approach**

**Advantages**

1. It significantly reduces the size of candidate sets using the Apriori principle.
2. It uses large itemset property.
3. It is easily parallelized.
4. It is easy to implement with all kind of real datasets.
5. The Apriori Algorithm calculates more sets of frequent items.

**Disadvantages**

1. It generates huge number of candidate sets.

2. When the longest frequent itemsets is *k*, Apriori needs *k* passes of database scans. So it will have low efficiency.
3. Repeatedly scanning the database and checking the candidates by pattern matching.
4. The computation time is very intensive at generating the candidate itemsets and computing the support values for application with very low support and vast amount of items.
5. The candidate generation could be extremely slow (pairs, triplets, etc.).
6. The candidate generation could generate duplicates depending on the implementation.
7. The counting method iterates through all of the transactions each time.
8. Constant items make the algorithm a lot heavier.
9. Huge memory consumption

**B. Without Candidate Generation Approach**

***1). FP-Growth Algorithm:-***

In Data Mining the task of finding frequent pattern in large databases is very important and has been studied in large scale in the past few years. The FP-Growth Algorithm, proposed by Han in, is an efficient and scalable method for mining the complete set of frequent patterns by pattern fragment growth, using an extended prefix-tree structure for storing compressed and crucial information about frequent patterns named frequent-pattern tree. The popularity and efficiency of FP-Growth Algorithm contributes with many studies that propose variations to improve his performance. F-P-Growth simplifies all the problems present in apriori by using a structure called an FP-Tree. In an FP-Tree each node represents an item and it's current count, and each branch represents a different association [5].

The whole algorithm is divided in 5 simple steps. Here we have a simple example:
Our client is named Mario and here we have his transactions: $T_{Mario}$= [ [beer, bread, butter, milk] , [beer, milk, butter], [beer, milk, cheese] , [beer, butter, diapers, cheese] , [beer, cheese, bread] ]

Step 1:

The first step is we count all the items in all the transactions $T_{Mario}$= [ beer: 5, bread: 2, butter: 3, milk: 3, cheese: 3, diapers: 1]

Step 2:

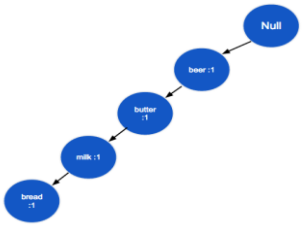
Next we apply the threshold we had set previously. For this example let's say we have a threshold of 30% so each item has to appear atleastt wice.
$T_{Mario}$= [ beer: 5, bread: 2, butter: 3, milk: 3, cheese: 3, ~~diapers: 1~~]

Step 3:

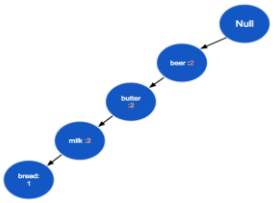
Now we sort the list according to the count of each item. $T_{MarioSorted}$ = [ beer: 5, butter: 3, milk: 3, cheese: 3, bread: 2]

Step 4:

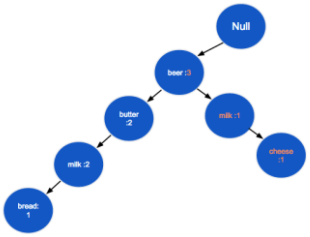
Now we build the tree. We go through each of the transactions and add all the items in the order they appear in our sorted list .**Transaction to add= [beer, bread, butter, milk]**



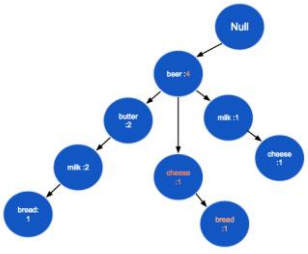

**Transaction 2: [beer, milk, butter]**




**Transaction3=[beer, milk, cheese]**

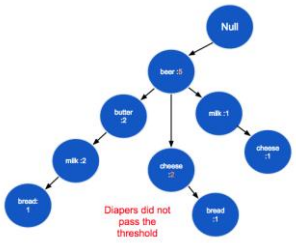



**Transaction 4=[beer, cheese, bread]**



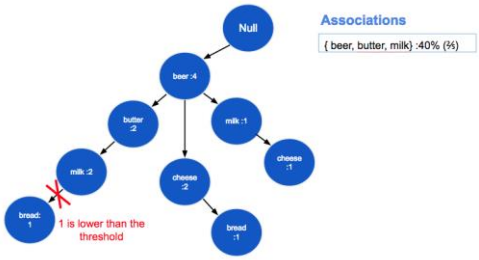

 

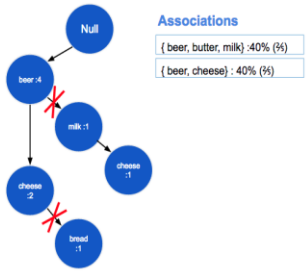
**Transaction 5=[beer, cheese, diapers]**




Step 5:

In order to get the associations now we go through every branch of the tree and only include in the association all the nodes whose count passed the threshold.




## 2). Advantages and Disadvantages of Without Candidate Generation Approach

**Advantages**

1. It does not break a long pattern of transaction.
2. It conserves complete information for frequent pattern mining.
3. It reduces irrelevant information or infrequent items are gone.
4. The frequency descending ordering is more likely to be shared.
5. It does not make transaction set larger than the original database.
6. It is much faster than Apriori algorithm.

**Disadvantages**

1. Frequent pattern tree may not fit in memory.
2. Frequent pattern tree is expensive to build. The time takes to build, but once it is built, frequent itemsets are read of easily.
3. If support is high, time is wasted, as the only pruning that can be done is on single items.
4. The support can only be calculated once the entire dataset is added to the FP-Tree.

## C. Comparisons between Apriori, Improved Apriori Algorithm and FB Growth

Association Rule Mining has attracted a lot of intention in research area of Data Mining and generation of association rules is completely dependent on finding Frequent Item sets. Various algorithms are.

| S.NO. | ALGORITHMS | TECHNIQUES | BENEFITS |
|---|---|---|---|
| 1. | Apriori | -Temporary tables for scanning<br>-logrithimic decoding | -Low system overhead and goodoperating performance<br>-efficiency higher than apriori algorithm |
| 2. | Improved apriori | -Variable Size Of Transaction on the basis of which transactions are reduced | -reduced the I/O cost<br>-reduced the size of Candidate Itemsets(CK) |
| 3. | FP growth tree | -Combine the apriori and fp tree structure of FP growth algo | -It doesnot generate conditional and sub-conditional patterns of the tree recursively<br>-it works faster than apriori for large database available for this purpose |

## V. Conclusion And Future

Association rule mining is utilized to find the much of the time happening designs in the database. Apriori calculation can be considered as one of the most seasoned calculation in the field of affiliation manage mining. This paper incorporate a concise outline of apriori calculation and late upgrades done in the range of apriori calculation. With the study on different enhanced calculations, it is presumed that the real concentration is to create less applicant sets which contains visit things inside a sensible measure of time. Likewise, in future some more calculations can be produced that requires just single output for the database and are proficient for expansive databases.

## VI. References

[1]. S. Paul, "*An Optimized Distributed Association Rule Mining Algorithm In Parallel And Distributed Data Mining With XML Data For Improved Response Time*", International Journal of Computer Science and Information Technology, Volume 2, Number 2, April 2010.

[2]. M.N. Moreno, S. Segrera and V.F. López, "*Association Rules: Problems", Solutions and new application Universidad de Salamanca*, Plaza Merced S/N, 37008, Salamanca.
[3]. K.P. Kumar and S. Arumugaperumal, "*Association Rule Mining and Medical Application; A Detailed Survey*", International Journal of Computer Application(0975-8887), Volume 80, number 17, October 2013.
[4]. E. Bala Krishna, B. Rama, A. Nagaraju, "*A Survey on Effective Mining of Negative Association Rules from Huge Databases*", International Journal of Computer Sciences and Engineering, Vol.3, Issue.9, pp-220-223, 2015.
[5]. V. Kavi, D. Joshi , "*A Survey on Enhancing Data Processing of Positive and Negative Association Rule Mining*", International Journal of Computer Sciences and Engineering, Vol.2, Issue.3, pp.139-143, 2014.
[6]. C. Wang, R. Li, and M. Fan, "*Mining Positively Correlated FrequentItemsets*," *Computer Applications*, vol. 27, pp. 108-109, 2007
[7]. J. Pei, J. Han, and H. Lu, "*Hmine: Hyper-structure mining of frequent patterns in large databases",* In ICDM, 2001, pp441–448.
[8]. N. Sethi, P. Sharma, "*Mining Frequent Pattern from Large Dynamic Database Using Compacting Data Sets*", International Journal of Scientific Research in Computer Science and Engineering, Vol.1, Issue.3, pp.31-34, 2013.
[9]. R. Trikha, J. Singh, "*Improving the efficiency of apriori algorithm by adding new parameters*", International Journal for Multi-Disciplinary Engineering and Business Management, Volume-2, Issue-2, June-2014
[10]. M. Al-Maolegi, B. Arkok, "*An improved apriori algorithm for association rules*", International Journal on Natural Language Computing (IJNLC) Vol. 3, No.1, February 2014

.

## Authors Profile

**Ms. Moksha Shridhar** pursed Bachelor of Engineering from ITM GOI in 2014. She is Currently pursuing her Master of Engineering from Madhav Institute of Technology and Science, Gwalior.

**Mr.Mahesh Parmar** as an Assistant Professor in CSE Dept. in MITS Gwalior and having 8 years of Academic and Professional experience. He received M.E. degree in Computer Engineering from SGSITS Indore in July 2010. His other qualifications are B.E. (Computer Science and Engineering, 2006). His area of expertise is Data Mining and Image Processing. He has published 15 research papers in International Journals and Conferences. He has also published 02 book chapters**.**