GLOBAL JOURNAL OF ENGINEERING SCIENCE AND RESEARCHES

A LITERATURE SURVEY ON FREQUENT ITEMSET MINING – AN ARM PERSPECTIVE

Kalaiyarasi. P[*1], Prof. Manikandan. M[2]
PG Scholar, Dept of CSE, Adhiyamaan College of Engineering, Hosur, India[*1]
Assistant Professor Dept of CSE, Adhiyamaan College of Engineering, Hosur, India[2]

ABSTRACT

Association Rules mining (ARM) which finds the relationship between distinct item sets plays an essential role in Item set mining. Frequent item set mining is one of the popular data mining techniques and it can be used in many data mining fields for finding highly correlated itemsets. Frequent items are those items that have been frequently used in the database. Infrequent itemset mining which is the inverse of frequent item set mining that finds the rarely occurring itemsets in the database. Several techniques were existing for mining frequent itemsets and infrequent itemsets with high computing time and are less scalable when the database size increases. This paper focuses on relating the existing algorithms that mines the frequent and infrequent itemsets which creates future researchers to find a way in the domain of association rule mining.

Keywords—Association Rules mining (ARM), Apriori, Frequent items, FP-growth, Infrequent Items, performance.

I. INTRODUCTION

In Data Mining, Association Rule mining (ARM)[1] is one of the popular technique used to find the correlation between the data items in the database based on some statistical measures but not considering the interesting of the business users. ARM is one of the oldest techniques in data mining. The goal of ARM is to find the relationship, correlation among different data sets in the database. Frequent itemset mining is an exploratory data mining technique widely used for discovering valuable correlations among data. Frequent itemsets mining is a core component of data mining and variations of association analysis, like association rule mining. Frequent items are those items that have been frequently used in the database. Infrequent itemset mining which is the inverse of frequent itemset mining that finds the rarely occurring itemsets in the database. Infrequent itemsets are produced from very big or huge data sets by applying some rules or association rule mining algorithms like Apriori technique, that take larger computing time to compute all the frequent itemsets. Extraction of frequent itemsets is a core step in many association analysis techniques. Itemsets that occurs rarely in the Database. Infrequent weighted Itemset mining finds application in the areas of Fraud Detection, Market Basket analysis. An item set is said to be Infrequent itemset if the total weight of the item is less than MIN -Threshold Value.

The frequent occurrence of item is expressed in terms of the support count. However, significantly less attention has been paid to mining of infrequent itemsets, but it has acquired significant usage in mining of negative association rules from infrequent itemset, fraud detection where rare patterns in financial or tax data may suggest unusual activity associated with fraudulent behavior, market basket analysis and in Bioinformatics where rare patterns in micro array data may suggest genetic disorders. Several frequent item set mining including Apriori, FP-Growth algorithm[2], AFOPT algorithm, NONORDFP algorithm, FP_Growth* algorithm, Broglet's FP-Growth, DynFP-Growth algorithm, Enhanced FP-Growth algorithm, IFP_min Algorithm and Transaction mapping algorithm were proposed.

II. LITERATURE SURVEY

A. Mining Frequent ItemSets without Candidate Generation

In the year 2000, Jiawei Han, jianPei and Yiwen Yin[3] explained about the construction of Frequent Pattern-tree data structure. Frequent Pattern-tree structure is an extended prefix tree structure for storing compressed, crucial information about frequent patterns and developed an efficient FP-tree based mining method, a complete set of Frequent Pattern-growth algorithm for mining frequent patterns. FP-Growth approach is based on divide and conquers strategy for producing the frequent item sets.

Step1: It firstly compresses the database showing frequent item set in to FP-tree. FP-tree is built using 2 passes over the data-set.

Step2: It divides the FP-tree in to a set of conditional database and mines each database separately, thus extract frequent item sets from FP-tree directly.

It consists of one root labeled as null, a set of item prefix sub trees as the children of the root, and a frequent .item header table. Each node in the item prefix sub tree consists of three fields: item-name, count and node link where item-name registers which item the node represents; count registers the number of transactions represented by the portion of path reaching this node, node link links to the next node in the FP- tree. Each item in the header table consists of two field item name and head of node link, which points to the first node in the FP-tree carrying the item name. FP-growth is mainly used for mining frequent item sets without candidate generation.

B. Efficient Mining of Weighted Association Rules

In the year 2000, Wei Wang, Jing Yang, Philip S. Yu discussed about the importance of weighted association rules [4]. It extends the tradition association rule problem by allowing a weight to be associated with each item in a transaction, to react interest/intensity of the item within the transaction. It provides us in turn with an opportunity to associate a weight parameter with each item in the resulting association rule. It is weighted association rule (WAR). WAR not only improves the confidence of the rules, but also provides a mechanism to do more effective target marketing by identifying or segmenting customers based on their potential degree of loyalty or volume of purchases. This approach mines WARs by ignoring the weight and finding the frequent itemsets through traditional frequent itemset discovery algorithm and is followed by introducing the weight during the rule generation. It not only results in shorter average execution times, but also produces higher quality results than the generalization of previous known methods on quantitative association rules.
Calculation of Weighted Item set:

Total weight(i)=weight(i) occurrence(i)*

C. Data Structure for Association Rule Mining using T-trees and P-trees

In the year 2004, Frans Coenen, Paul Leng, and Shakil Ahmed introduced about T-Trees and P-Trees Data structures for Association Rule Mining[5]. It explains about efficient data storage mechanism for itemset storage, the T-tree, is described. It also considers data Pre-processing and describes the P-tree, which is used to perform a partial computation of support totals. It shows use of these structures offers significant advantages with respect to existing ARM techniques.

(i) Total-Support Tree:

The Total-Support Tree Data structure can be optimized by storing levels in the tree in the form of arrays, thus reducing the number of links needed and providing direct indexing. It is more convenient to build a “reverse” version of the tree, refers to compressed set enumeration tree as a T-tree (Total support tree), where each node in the T-tree is an object (T-tree Node) comprised of a support value (sup) and a reference (child Reference) to an array of child T-tree nodes. It uses Apriori algorithm to generate the combinations.

(ii) Partial-Support Tree:

The disadvantage of Apriori is that the same records are repeatedly reexamined. Here it introduces the concept of partial support counting using the P-tree (Partial support tree). The idea is to copy the input data (in one pass) into a data structure, which maintains all the relevant aspects of the input, and then mine this structure. In this respect, the P-tree offers two advantages: It merges duplicated records and records with common leading substrings, thus reducing the storage and processing requirements for these and it allows partial counts of the support for individual nodes within the tree to be accumulated effectively as the tree is constructed.

To construct a P-tree, pass through the input data record by record. P-tree will contain all the itemsets present as distinct records in the input data. The sup stored at each node is an incomplete support total, comprised of the sum of the supports stored in the sub tree of the node, which derives from all its lexicographically sets has been included.

D. Fast Algorithms for Frequent Item Set Mining

In the year 2005, Go sta Grahne and Jianfei Zhu[6] explained about how to construct frequent pattern tree by using Fast Algorithms to mine the Infrequent Item Sets. Efficient algorithms for mining frequent itemsets are crucial for mining association rules as well as for many other data mining tasks. Methods for mining frequent itemsets have been implemented using a prefix-tree structure, known as an FP-tree, for storing compressed information about frequent itemsets. It introduced a novel FP-array technique that greatly reduces the need to traverse FP-trees, thus obtaining significantly improved performance for FP-tree-based algorithms. This technique works especially well for sparse data sets. Furthermore, new algorithms for mining all, maximal, and closed frequent itemsets. These algorithms use the frequent pattern tree data structure in combination with the FP-array technique efficiently and incorporate various optimization techniques. Even though the algorithms consume much memory when the data sets are sparse, they are still the fastest ones when the minimum support is low. Moreover, they are always among the fastest algorithms and consume less memory than other methods when the data sets are dense.

E. A Transactional Mapping Algorithm for Frequent Item Set Mining

In the year 2007, Mingjun Song and Sanguthevar Rajasekaran presented Transaction Mapping algorithm[7] for mining complete frequent itemsets. In this algorithm, transaction ids of each itemset are mapped and compressed to continuous transaction intervals in a different space and the counting of itemsets is performed by intersecting these interval lists in a depth-first order along the lexicographic tree. When the compression coefficient becomes smaller than the average number of comparisons for intervals intersection at a certain level, the algorithm switches to transaction id intersection.

Algorithm: /* Transactional Mapping algorithm */

Input: Transactional database.

Output: all infrequent item sets

```
[1] Construct the transaction tree with the count for each node.
[2] Construct the transaction interval lists.
[3] Construct the lexicographic tree in a depth first order.
[4] create the header table with the fields.
[5] For each transaction create a path as sorted in lexicographic order.
```

F . On Minimal Infrequent Item Set Mining

In the year 2007, David J. Haglin and Anna M. Manning introduced a new algorithm for mining the minimal infrequent item sets[8]. This algorithm is designed specifically for finding these rare itemsets. This algorithm can be adapted to handle the more traditional dataset definition and to handle finding minimal infrequent itemsets. Initially, a ranking of items is prepared by computing the support of each of the items and then creating a list of items in ascending order of support. Minimal infrequent itemsets are discovered by considering each item in rank order, on the support set of the dataset with respect to item considering only those items with higher rank and then checking each candidate minimal infrequent items against the original dataset. This considers only higher-ranking items in the recursion is to maintain a vector indicating which items remain viable at each level of the recursion. It would also be useful to find other pruning strategies to improve the running time and memory space requirements.

Given a dataset D and an integer threshold τ, an itemset I is :
τ –occur-rent if $|D\ (I)| = \tau$

τ -frequent if |D (I)| ≥ τ
τ -infrequent if |D (I)| < τ.

G. Minimally Infrequent Item Set Mining using Pattern Growth Paradigm and Residual Tree

In the year 2012, Ashish Gupta, Akshay Mittal, Arnab Bhattacharya identified the concept of minimally infrequent itemsets. A minimally infrequent itemset has no subset which is also infrequent. It introduces the concept of residual trees [9], where different thresholds are used for finding frequent itemsets for different lengths of the itemset. Frequent itemset mining algorithms can be broadly classified into two categories: candidate generation and test paradigm and Pattern-growth paradigm.

The pattern-growth paradigm is to propose an algorithm IFP min (Infrequent Pattern min) for mining minimally infrequent itemsets. For some datasets, the set of infrequent itemsets can be exponentially large. Reporting an infrequent itemset which has an infrequent proper subset is redundant; hence, it is essential to report only the minimally infrequent itemsets. If the support threshold is too high, then less number of frequent itemsets will be generated resulting in loss of valuable association rules. On the other hand, when the support threshold is too low, a large number of frequent itemsets and consequently large number of association rules are generated, thereby making it difficult for the user to choose the important ones. A part of the problem lies in the fact that a single threshold is used for generating frequent itemsets irrespective of the length of the itemset. To overcome this problem, Multiple Level Minimum Support (MLMS) model was proposed, where separate thresholds are assigned to itemsets of different sizes in order to constrain the number of frequent itemsets mined.

H. Efficient Mining of Frequent Item Sets on Large Uncertain Databases

In the year 2012, Liang Wang, David Wai-Lok Cheung, Reynold Cheng discussed about the problem of extracting frequent item sets from a large uncertain database[10], interpreted under the Possible World Semantics (PWS).This issue is technically challenging, since an uncertain database contains an exponential number of possible worlds. By observing that the mining process can be modeled as a Poisson binomial distribution, developed an approximate algorithm, which can efficiently and accurately discover frequent item sets in a large uncertain database and also discussed about important issue of maintaining the mining result for a database that is evolving.

Incremental mining algorithms enables the probabilistic Frequent Itemset (PFI) and reduces the need of re-executing the whole mining algorithm on the new database which is often more expensive and unnecessary. These approaches support both tuple and attribute uncertainty, which are two common uncertain database models. Also performed extensive evaluation on real and synthetic data sets to validate our approaches.
The probability can be calculated by using Bayes' Theorem,

$$\boxed{P(H|X) = \frac{P(X|H)\,P(H)}{P(X)}}$$

I. Mining Association Rules between Sets of Items in Large Databases

In the year 2013, A. Krishna Kumar, D. Amrita, N. Swathi Priya, suggested mining of association rules from large data sets [11]. A top-down progressive deepening method is developed for efficient mining of multiple-level association rules from large transaction databases based on the Apriori principle. This is method is for mining multiple-level association rules is introduced, which uses a hierarchy-information encoded transaction table instead of the original transaction table. In data mining, query is usually relevant to only a portion of the transaction database, instead all the items. It has the benefit to first collect the relevant set of data and then work repeatedly on the task-relevant set. Encoding can be performed during the collection of task relevant data and, thus, there is no extra encoding pass required. An encoded string, which represents a position in a hierarchy, requires fewer bits than the corresponding object identifier or bar-code. Therefore, it is often beneficial to use an encoded table, although our method does not rely on the derivation of such an encoded table because the encoding can always be performed.

III. CONCLUSION

Weighted Item set mining is an exploratory information mining system generally utilized for uncovering profitable connections among information. The main benefit is to perform Infrequent item set mining is to improve the rarely occurred datasets in the transactions. Several frequent itemset mining algorithms such as Apriori to FP- growth were exist. Our research article focuses on providing a review on all the association rule mining algorithms and hence it acts as guide for future researchers in Itemset mining.

REFERENCES

[1] Agrawal R , Imieliński T , Swami A.”Mining association rules between sets of items in large databases”.In proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, pages 207-216, Washington, DC, 1993.

[2] Sakthi Nathiarasan A, Kalaiyarasi P, Manikandan M, “Literature Review on Infrequent Itemset Mining Algorithms”, International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), ISSN:2271-1021, Volume 3, Issue 8, August 2014.

[3] Jiawei Han, Jian Pei and Yiwen Yin, ‘Mining Frequent Patterns without Candidate Generation’, Proceedings of ACM SIGMOD International Conference Management of Data, pp. 1-12, 2000.

[4] Wei Wang, Jiong Yang, Philip S.Yu, ‘Efficient Mining of Weighted Association Rules(WAR)’, Proceedings of 6th ACM SIGKDD Int’l Conf. Knowledge Discovery and data Mining (KDD’00), pp. 270-274, 2000.

[5] Frans Coenen, Paul Leng & Shakil Ahmed, ‘Data Structure for Association Rule Mining: T-Trees and P-Trees’, IEEE Transactions on Knowledge and Data Egineering, vol.16, no.6, 2004.

[6] Go sta Grahne & Jianfei Zhu, ‘Fast Algorithms for Frequent Itemset Mining Using FP-Trees’, IEEE Transactions on Knowledge and Data engineering, vol.17, no.10, 2005.

[7] David J. Haglin and Anna M. Manning, ‘On Minimal Infrequent Itemset Mining’, Proceedings of International Conference Data Mining (DMIN ’07), pp. 141-147, 2007.

[8] Mingjun Song, Sanguthevar Rajasekaran, 2006 ‘A Transaction Mapping Algorithm for Frequent Itemsets Mining’, IEEE transactions on knowledge and data engineering, vol.18, no. 4, april 2006.

[9] Liang Wang, David Wai-Lok Cheung, Reynold Cheng, ‘Efficient Mining of frequent item sets on Large uncertain databases’, IEEE transactions on knowledge and data engineering, vol. 24, no. 12, 2012.

[10] Ashish Gupta, Akshay Mittal & Arnab Bhattacharya, 2011 ‘Minimally Infrequent Item set mining using Pattern growth paradigm and Residual Tree’, Proceedings of International Conference Management of Data (COMAD), pp. 57-68.

[11] KrishnaKumar A, Amrita D & SwathiPriya N, ‘Mining Association Rules between Sets of Items in Large Databases’, International Journal of Science and Modern Engineering (IJISME)ISSN: 2319-6386,Volume-1, Issue-5, 2013.