A Survey on Moving Towards Frequent Pattern Growth for Infrequent Weighted Itemset Mining

Miss. Rituja M. Zagade
Computer Engineering Department,JSPM,NTC
RSSOER,Savitribai Phule Pune University
Pune,India
ritujazagade55@gmail.com

Prof. Megha V. Borole
Assistant Professor,Computer Engineering Department,
JSPM,NTC,RSSOER,Savitribai Phule Pune University
Pune,India
megha.borole@gmail.com

***Abstract*—** Data Mining and knowledge discovery is one of the important areas. In this paper we are presenting a survey on various methods for frequent pattern mining. From the past decade, frequent pattern mining plays a very important role but it does not consider the weight factor or value of the items. The very first and basic technique to find the correlation of data is Association Rule Mining. In ARM we have mainly two concepts that is support and confidence. The frequent itemsets are items that come in a data set frequently that is occurrence of that item again and again in that dataset. The new concept is to find frequent itemsets along with its weight that is weighted itemset mining. In this paper we are presenting literature survey of various frequent pattern mining and weighted itemset mining**.** This paper has different techniques related to frequent and weighted infrequent itemset mining. This paper we study the of various Existing Algorithms related to frequent and infrequent itemset mining which creates a path for future researches in the field of Association Rule Mining.

Keywords- *Data Mining, frequent pattern Mining, itemset mining, infrequent weighted itemset.*

.

*****

I. Introduction

Overview generally, data mining also called as data or knowledge discovery is the process of analyzing data from different perspectives and summarizing it into useful information - information that can be used to increase revenue, cuts costs, or both. Data mining software is one of a number of analytical tools for analyzing data. It allows users to analyze data from many different dimensions or angles, categorize it, and summarize the relationships identified. Technically, data mining is the process of finding correlations or patterns among dozens of fields in large relational databases. 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 technique in data mining. The goal of ARM is to find the relationship, correlation among different data sets in the database. 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. 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. 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 microarray data may suggest genetic disorders. Several frequent item set mining including Apriori, FP-Growth algorithm, FP-GROWTH* algorithm. This paper tackles the issue of discovering rare and weighted itemsets, i.e., the infrequent weighted itemset (IWI) mining problem, two algorithms that perform IWI and Minimal IWI mining efficiently.

II. Literature Survey

a. Review:

A. The Apriori Algorithm

Apriori[2] was the very first proposed algorithm in association rule mining, to identify the frequent itemsets in the large transactional database. Apriori is one of the old algorithm which is used in many data mining applications. It works in two phases. In the first phase it generates all possible Itemsets combinations. In this combinations will act as possible candidates. Candidate set is generated in the first phase, which will be used in subsequent phases. Apriori algorithm works mainly on two concepts support and confidence. First the minimum support is applied to find all frequent itemsets in a database and Second, these frequent itemsets and the minimum confidence constraint are used to form rules.

Apriori Algorithm:

```
Procedure Apriori (T, minSupport)
{
//T is the database and minSupport is the minimum
support
L1= {frequent items};
for (k= 2; Lk-1 !=∅; k++)
{
Ck= candidates generated from Lk-1
for each transaction t in database
do
{
#increment the count of all candidates in Ck that are
contained in t
Lk = candidates in Ck with minSupport
}//end for each
}//end for
return∪k Lk ;
}
```

The main disadvantage of Apriori is the generation of large number of candidate sets. Because of this it takes so much time to find itemsets, efficiency of apriori can be improved by Monotonicity property, hash based technique, Partioning methods and more.

B. FP-growth Algorithm

To overcome the disadvantages of Apriori can be improved by Frequent pattern Growth algorithm [3]. This algorithm is work without generating the candidate sets. FP-growth algorithm proposes a tree structure called FP tree structure, going to collect information from the database and creates an optimized data structure as Conditional pattern. First, it Scans the transaction database DB once and Collects the set of frequent items F and their supports and then Sort the frequent itemsets in descending order as L, based on the support count. This algorithm reduces the number of candidate set generation, number of transactions, number of comparisons .Hence, as compared with Apriori it gives improved efficiency.

Algorithm FP-growth:

Input:
-A transactional database *DB* and a minimum support threshold ξ.

Output:
- frequent pattern tree, FP-tree

/*phase1: */
(1)It Scan the transactional database.
(2) Collect the set of frequent items *F* and their supports.
Sort *F* in support descending order as *L*. /* the *list* of frequent items*/
(3) [3] Create the root of an FP-tree, *T*, and label it as "root" /* for each transaction do */
(4) Select and sort the frequent items in *Trans* according to the order of *L*.
(5) perform the insert-tree function /* call insert-tree function recursively */ /*phase2: */

Input:
An FP-tree constructed in the above algorithm,
D – transaction database;
s – minimum support threshold.

Output:
-The complete set of frequent patterns.
(1) call the FP-Growth function
(2) check if the tree has a single path,
(3) then for each combination (denoted as *B*) of the nodes in the path *P* do
(4) generate pattern B ∪ A with support=minimum support of nodes in B
(5) else
(6) construct B's conditional pattern base and B's conditional FP-tree

c. FP-Growth Algorithm*

Grahne et al [6], in his found that FP-trees uses 80% of CPU utilization for traversing the trees. Fpgrowth* algorithm uses FP-tree data structure in combination with the array-based and incorporates various optimization techniques. To reduce the traversal time of FP-tree Array-based technique is used. It reduces the memory utilization compared to FP-growth Algorithm.

b. Related Work:

Paper Title- "Minimally Infrequent Item set Mining using Pattern-Growth Paradigm and Residual Trees"

This paper contains a new algorithm pattern growth paradigm to find minimally infrequent itemsets. A smallest amount of infrequent itemset has no subset which is also infrequent. In this the novel concept of residual trees also has attention. Furthermore utilize the residual trees to mine multiple level minimum support itemsets where different thresholds are used for finding frequent itemsets for different lengths of the itemset. In this paper there is pattern-growth paradigm to has a algorithm IFP min for mining rarely found infrequent itemsets. Some dataset has huge amount of the set of infrequent itemsets. This cause to an infrequent itemset which has an infrequent proper subset is redundant. Therefore it is must that assist only the minimally infrequent itemsets.

Advantages

- There is use of threshold to find out frequent item sets for different lengths of the item set.
- In web mining the infrequent Itemset mining has shown its utility.
- It is used to find minimally infrequent itemsets.

Disadvantages

- If threshold is high then less number of infrequent item sets are generated.
- Large number of infrequent item sets is generated when the threshold is too low.

Paper Title- "Mining Weighted Association Rules without Pre assigned Weights"

Association Rule Mining has main two concepts support and confidence. Where, support is used for mining the weighted association rules. This method not required the preassigned weights in the transactions. To find the quality link based models are used in the transaction. Fast mining algorithm is also used. Association rule mining used to find correlation in the transactional databases. Association rule mining that help us to uncover relationships between seemingly unrelated data in a transactional database. It has many applications like cross marketing, classification, text mining, Web log analysis, and recommendation systems. The classical model of association rule mining has the support measure, which treats every transaction equally. In contrast, different transactions have different weights in real-life data sets. For example, in the market basket data, each transaction is recorded with some profit.

Advantages

- Association rule mining is used to find frequent itemsets in transaction databases for association rules.
- The link based models are used to find the quality of the transaction.

Disadvantages

- The frequent item set in the transactional dataset may not be important.
- It not consider the weight of the itemsets.

Paper Title- “On Minimal Infrequent Item set Mining”

Association rule mining is used to find frequent itemsets in transaction databases for association rules. which is used to find the implicit relationships among the data. It has very important role in data mining and has lots of practical applications like cross marketing, classification, text mining, Web log analysis, and recommendation systems A minimally infrequent item set has no subset which is also infrequent. We also introduce the novel concept of residual trees. We further utilize the residual trees to mine multiple level minimum support item sets where different thresholds are use for finding frequent item sets for different lengths of the item set. Finally, we analyze the behavior of our algorithm with respect to different parameters and show through experiments that it outperforms the competing ones. This paper conations the pattern-growth paradigm to propose an algorithm IFP min for mining minimally infrequent item sets.

Advantages

- The infrequent Itemset mining has demonstrated its utility in web mining.
- Association rule mining used to find large transaction databases for association rules.

Disadvantages

- Support of each item in the dataset is not accurate.
- Item appearing in every transaction cannot be part of any MII.

Paper Title- “Probabilistic Frequent Item set Mining in Uncertain Databases”

Probabilistic frequent itemset mining differs from traditional techniques applied to standard “certain” transaction databases. This paper focused on a new probabilistic formulation of frequent item sets based on possible world semantics. In this probabilistic context, an item set X is called frequent if the probability that X occurs in at least minSup transactions is above a given threshold τ. According to our survey this is the first approach addressing this problem under possible word semantics. In this we have probabilistic formulations which present a framework which is able to solve the Probabilistic Frequent Itemset Mining (PFIM) problem efficiently. In this paper gives the experimental result which shows orders of magnitude faster than straight-forward approaches.

Advantages

- A probabilistic formulation of frequent item sets is based on possible world semantics of the database.
- This scheme is to find the Probabilistic Frequent Itemset Mining (PFIM) problem efficiently.

Disadvantages

- It not gives the accurate item set identity whether an item or item set is frequent in the dataset.
- They do not consider item sets with more than one item.

Paper Title- “Weighted Association Rule Mining using Weighted Support and Significance Framework”

This paper focuses on the discovering significant binary relationships in transaction datasets in a weighted item. Conventional model of association rule mining is adapted to handle weighted association rule mining problems where each item is allowed to have a weight. In this main goal is to find the significant relationships involving items with significant weights. The problem of invalidation of the “downward closure property” in the weighted setting is solved by using an improved model of weighted support measurements and exploiting a “weighted downward closure property”. A new algorithm called Weighted Association Rule Mining (WARM) is developed based on the improved model. This algorithm is efficient and scalable in discovering significant relationships in weighted items.

Advantages

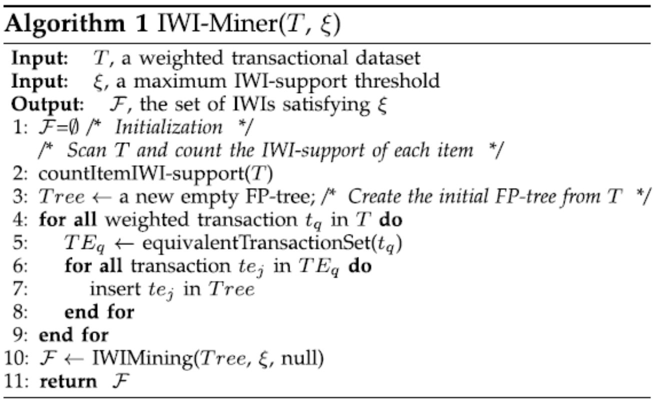
- System used to steer the mining focus to those significant relationships involving items with significant weights.
- This system is efficient in finding out both scalable and efficient to find the weighted itemsets in the database.

Disadvantages

- This technique cannot discover rare item rules without causing frequent items to generate too many unnecessary rules.
- This problem is not solved by the sorted closure property.

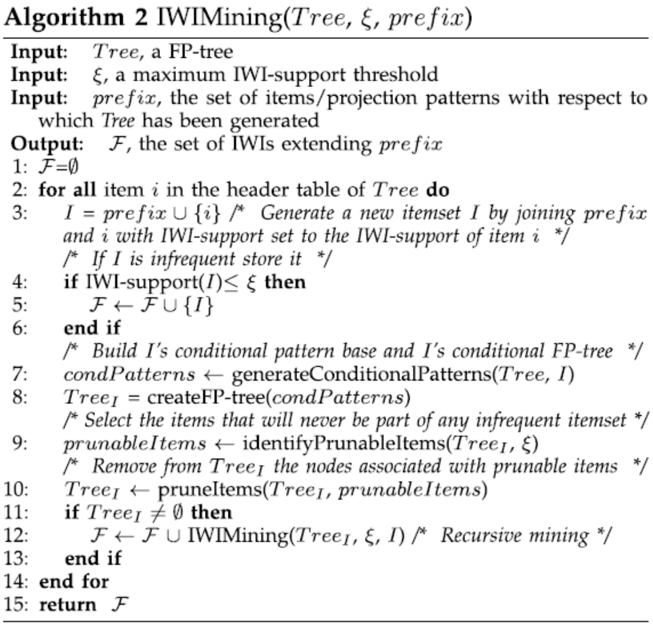
III. Algorithms/Methodology

IWI Miner and MIWI Miner algorithms are used.



Algorithm 1 IWI-Miner(T, ξ)

```
Input:  T, a weighted transactional dataset
Input:  ξ, a maximum IWI-support threshold
Output: F, the set of IWIs satisfying ξ
1: F=∅ /* Initialization */
   /* Scan T and count the IWI-support of each item */
2: countItemIWI-support(T)
3: Tree ← a new empty FP-tree; /* Create the initial FP-tree from T */
4: for all weighted transaction t_q in T do
5:    TE_q ← equivalentTransactionSet(t_q)
6:    for all transaction te_j in TE_q do
7:       insert te_j in Tree
8:    end for
9: end for
10: F ← IWIMining(Tree, ξ, null)
11: return F
```



Algorithm 2 IWIMining($Tree$, ξ, $prefix$)

```
Input:  Tree, a FP-tree
Input:  ξ, a maximum IWI-support threshold
Input:  prefix, the set of items/projection patterns with respect to
        which Tree has been generated
Output: F, the set of IWIs extending prefix
1: F=∅
2: for all item i in the header table of Tree do
3:    I = prefix ∪ {i} /* Generate a new itemset I by joining prefix
      and i with IWI-support set to the IWI-support of item i */
      /* If I is infrequent store it */
4:    if IWI-support(I) ≤ ξ then
5:       F ← F ∪ {I}
6:    end if
      /* Build I's conditional pattern base and I's conditional FP-tree */
7:    condPatterns ← generateConditionalPatterns(Tree, I)
8:    Tree_I = createFP-tree(condPatterns)
      /* Select the items that will never be part of any infrequent itemset */
9:    prunableItems ← identifyPrunableItems(Tree_I, ξ)
      /* Remove from Tree_I the nodes associated with prunable items */
10:   Tree_I ← pruneItems(Tree_I, prunableItems)
11:   if Tree_I ≠ ∅ then
12:      F ← F ∪ IWIMining(Tree_I, ξ, I) /* Recursive mining */
13:   end if
14: end for
15: return F
```

Algorithm 1 is IWI Miner pseudocode is represented. The first steps i.e. lines 2-9 of Algorithm 1 is used to generate the FP-tree associated with the input weighted data set T. Then, at line 10 the recursive mining process is invoked on the constructed FP-tree. The FP-tree is initially occupy with the set of equivalent transactions generated from T. For each weighted transaction, the equivalent set is generated by applying function equivalent Transaction Set at line 5, which implements the transactional data set equivalence transformation. Once a compact FP-tree representation of the weighted data set T has been created, the recursive itemset mining process is executed (Algorithm 1, line 10). Algorithm 2 contains pseudocode of the mining procedure. IWI Miner depend on a projection based approach [10], items belonging to the header table associated with the input FP-tree are iteratively considered given in lines 2-14. Initially, at line 3 each item is combined with the current prefix to generate a new itemset I. in line 4-6 If I is infrequent, then it is stored in the output IWI set F. Then, at line 7-8 the FP-tree projected with respect to I is generated and the IWI Mining procedure is recursively applied on the projected tree to mine all infrequent extensions of I at line 12. Unlike conventional FP-Growth-like algorithms [10], IWI Miner adopts a different pruning strategy see lines 9-10 for detail. According to Property, identify Prunable Items procedure visits the FPtree and identifies items that are only included in paths whose leaves have an IWI-support above

ξ. since in line 10, they cannot be part of any IWI, they are pruned.

IV. CONCLUSION

This paper presents a study on various methods of frequent itemsets mining. Frequent itemsets mining in which we have main concept association rule mining which will be used to find relation between data for that purpose there are many algorithms proposed. Frequent itemsets mining with its weight is now emerging concept. Weighted itemset mining is used to find the profitable connection between the data. There are frequent and infrequent items in database. Infrequent itemsets are nothing but items which are rarely found in database. In this infrequent itemset mining the main advantage was to improve the profit of rarely found datasets in the transactions database. In this, first is to find the frequent item set mining that is find all frequent itemsets and then discover the infrequent weighted item sets. Some, frequent itemset mining algorithms as Apriori to FPgrowth are there but in that generation of candidate sets is large. As per our study of all the existing algorithms IWI is the most effective algorithm, which computes in very less computing time. It improves the efficiency of performance when the database is large, computes the weighted transaction too.

ACKNOWLEDGMENT

The authors would like to thanks the Department of Computer Engineering JSPM,NTC,Rajarshi Shahu School of Engineering & Research, Pune, India for the guidance and cooperation.

REFERENCES

[1] Agrawal R, Imielinski 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] R. Agrawal and R. Srikant, “Fast Algorithms for Mining Association Rules,” Proc. 20th Int’l Conf. Very Large Data Bases (VLDB ’94), pp. 487-499, 1994. Copyright to IJARCCE www.ijarcce.com 7673

[3] Luca Cagliero and Paolo Garza “Infrequent Weighted Itemset Mining using Frequent Pattern Growth”, IEEE Transactions on Knowledge and Data Engineering, pp. 1-14, 2013.

[4] Liu,G. , Lu ,H. , Yu ,J. X., Wang, W., & Xiao, X.. ”AFOPT:An Efficient Implementation of Pattern Growth Approach”, In Proc. IEEE ICDM'03 Workshop FIMI'03, 2003.

[5] BalazesRacz,” nonordfp: An FP-Growth Variation without Rebuilding the FP-Tree”, 2nd Int'l Workshop on Frequent Itemset Mining Implementations FIMI2004

[6] Grahne O. and Zhu J. “Efficiently Using Prefix-trees in Mining Frequent Itemsets”, In Proc. of the IEEE ICDM Workshop on Frequent Itemset Mining, 2004.

[7] Cornelia Gyorodi, Robert Gyorodi, T. Cofeey& S. Holban –”Mining association rules using Dynamic FP-trees” – în Proceedings of The Irish Signal and Systems Conference, University of Limerick, Limerick, Ireland, 30th June-2nd July 2003, ISBN 0-9542973-1-8, pag. 76-82.

[8] Grahne G. and Zhu J., “Efficiently Using Prefix-Trees in mining Frequent Item sets,” Proc. ICDM 2003Workshop Frequent Item set Mining Implementations, (2003).

[9] A.Gupta, A. Mittal, and A. Bhattacharya, “Minimally Infrequent Itemset Mining Using Pattern-Growth Paradigm and Residual Trees,” Proc. Int’l Conf. Management of Data (COMAD), pp. 57- 68, 2011.

[10] J. Han, J. Pei, and Y. Yin, ”Mining frequent patterns without candidate generation,” Procedings of ACM SIGMOD International Conference on Management of Data, ACM Press, Dallas, Texas, pp. 1-12, May 2000.

[11] Han, J. , Pei, J. , & Yin, Y. “Mining frequent patterns without candidate generation”. In Proc. ACM-SIGMOD Int. Conf. Management of Data (SIGMOD ’96), Page 205-216, 2000.