



Survey on Infrequent Weighted Itemset Mining Using FP Growth

M.Hamsathvani¹, D.Rajeswari², R.Kalaiselvi³

PG Scholar (M.E), Angel College of Engineering and Technology, Tiruppur, India¹

Assistant Professor, Angel College of Engineering and Technology, Tiruppur, India²

PG Scholar (M.E), Angel College of Engineering and Technology, Tiruppur, India³

ABSTRACT: A new algorithm for infrequent item set mining, finding infrequent weighted item set in transaction database. Frequent weighted item sets represent correlation regularly holding in data in which items may weight differently. The research society has focused on the infrequent weighted item set mining problem. Infrequent weighted item set discover item sets whose frequency of occurrence in the analyzed data is less than or equal to a maximum threshold. To discover infrequent weighted item set, two algorithms are discovered Infrequent weighted item set IWI and Minimal infrequent item set MIWI. In this survey is focused on the infrequent weighted item sets, from transactional weighted data sets to address IWI support measure is defined as a weighted frequency of occurrence of an item set in the analyzed data. Occurrence weights derived from the weights associated with items in each transaction and applying a given cost function.

KEYWORDS: clustering, classification, FP tree, FP growth, association rules, and data mining

I.INTRODUCTION

Data mining is known for discovering earlier, suitable, original, useful and reasonable patterns in large databases. Due to the availability of huge amount of data and the need to transform such data into useful information and knowledge, data mining has grown to be the most widely used technique in the society as a whole. Thus data mining [1] can be used for applications ranging from market analysis, fraud detection and customer retention, to production control and science exploration. Association rule mining [1] aims to look at large transaction databases for association rules, which may reveal the understood relationships among the data attributes. It has turned into a successful research topic in data mining and has numerous practical applications, including cross marketing, classification, text mining. The classical model of association rule mining employs the support measure, which treats every transaction equally. In contrast, different transactions have different weights in real-life data sets. Mining frequent item sets has found extensive utilization in various data mining applications including consumer market-basket analysis, inference of patterns from web page access logs. Research has been conducted in finding efficient algorithms for frequent item set mining, especially in finding association rules.

Weighted support [3] is used instead of support used in traditional pattern mining which was simply the count of occurrence of item sets in each transaction. Weighted support calculated by making use of the weights of items resulted in the selection of important patterns. An item set is significant if its weighted support is above a pre-defined minimum weighted support. Infrequent item set mining algorithms still suffer from their incapability to take local item interestingness into account during the mining phase. Infrequent item set considerably less attention has been remunerated to mining of infrequent item sets, but it has acquired major usage in mining of unconstructive association rules from infrequent item set, fraud detection, statistical disclosure risk assessment from census data, market basket analysis and bioinformatics. In this survey focused on various frequent item set and infrequent item set mining such as Apriori, FP tree, FP growth, mining infrequent item set, Infrequent weighted item set IWI and Minimal infrequent item set MIWI.



International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2014

II. REVIEW ON APRIORI ALGORITHM

Apriori algorithm for frequent item set mining and association rule over transactional database. It proceeds by identifying the frequent individual items in the database and extending them to larger and larger item sets as long as those item sets appear sufficiently often in the database. To identify the frequent item sets in the large transaction database. Two stages of Apriori algorithm [2] first stage count item occurrence and generate candidate item set and second count support candidate item.

Apriori Algorithm 1:

```
Il = {Input as item sets};
For (k = 2; Lk-1 ≠ 0; k++) do begin
Ck = apriori gen (Lk-1);
For all transactions t ∈ D do begin
Ct = sub set (Ck, t);
For all candidates c ∈ Ct do
C.count++;
End
Lk = {c ∈ Ck | c: count ≥ minsup}
end
Output = ∪k Lk;
```

III. REVIEW ON FREQUENT PATTERN TREE

Frequent pattern tree [4] (FP-tree) structure, which is an extended prefix-tree structure for storing compressed, crucial information about frequent patterns, and develop an efficient FP-tree-based mining method. Only frequent length-1 items will have nodes in the tree, and the tree nodes are arranged in such a way that more frequently occurring nodes will have better chances of sharing nodes than less frequently occurring ones.

FP-tree construction Algorithm 2:

Input: Transaction database and a minimum support threshold

Output: FP-tree

FP-tree is constructed in the following steps.

1. Collect the set of frequent items F and their supports.
2. Sort F in support descending order as L , the list of frequent items.
3. Create the root of an FP-tree, T , and label it as "null" .
4. Sort the frequent items in Trans according to the order of L .
5. Let the sorted frequent item list in Trans be $[p|P]$, where p is the first element and P is the remaining list.
6. Insert_{tree} ($[p|P]$, T).
7. $T \rightarrow N$. item-name = p . item-name

If T has a child N such that N . item-name = p . item-name, then increment N 's count by 1; else create a new node N , and let its count be 1, its parent link be linked to T , and its node-link be linked to the nodes with the same item-name via the node-link structure. If P is nonempty, call

Insert_{tree} (P , N) recursively

IV. REVIEW ON FP- GROWTH

FP-growth mining [4] the complete set of frequent patterns by pattern fragment growth. For a large database is compressed into a highly condensed, much smaller data structure, which avoids costly, repeated database scans and FP-tree-based mining adopts a pattern fragment growth method to avoid the costly generation of a large number of candidate sets, and a partitioning based, divide and conquer method is used to decompose the mining task into a set of smaller tasks for mining confined patterns in conditional databases, which dramatically reduces the search space.

FP- Growth Algorithm 3

Input: FP-tree Algorithm 2 and minimum support value

Output: The complete set of frequent patterns.

FP-growth (FP-tree, null)



International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2014

Procedure FP-growth (Tree; ∞)

1. If Tree contains a single path P Then for each combination (denoted as β) of the nodes in the path P do
2. Generate pattern $\beta \cup \infty$ with support = minimum support of nodes in β else for each a_i in the header of Tree
3. generate pattern $\beta = a_i \cup \infty$ with support = a_i : support;
4. Construct β 's conditional pattern base and then β 's conditional FP-tree Tree β ;

V.REVIEW ON MINIMALLY INFREQUENT ITEM SET

To find minimal infrequent item [5] sets developed for finding minimal unique item sets. Infrequent item set which has an infrequent proper subset is redundant, since the former can be deduced from the latter. i) Computing the support of each item, which is needed to produce a rank-ordering of the viable items by support ii) determining the viability of each item pruning some of the items from consideration iii) computing the support set of each viable item resulting in a memory efficient representation of the lists of TIDs for each support set

VI.REVIEW ON INFREQUENT WEIGHTED ITEM SET

IWI Miner is a FP-growth-like mining algorithm that performs projection-based item set mining. FP-growth mining steps:

1. FP-tree creation and (b)
2. Recursive item set mining from the FP tree index.
3. IWI Miner discovers infrequent weighted item sets instead of frequent (unweighted) ones.

Modifications with respect to FP-growth have been introduced:

- (i) A novel pruning strategy for pruning part of the search space early and
- (ii) a slightly modified FP tree structure, which allows storing the IWI-support value associated with each node.

Infrequent weighted item set Algorithm 4

Input- weighted transaction dataset and support value)

IWI (T, E)

- 1) F=0
 - 2) Count item IWI (T)
 - 3) Construct FP tree
 - 4) For all weighted transaction
 - 5) Calculate Equivalent transaction
 - 6) For all transaction create and insert into FP tree
- Output – Set of satisfying E

VII.REVIEW ON MINIMAL INFREQUENT ITEM SET

MIWI Miner focuses on generating only minimal infrequent patterns, the recursive extraction in the MIWI Mining procedure is stopped as soon as an infrequent item set occurs. It finds both the infrequent item sets and minimal infrequent item set mining.

IWI mining (T, E, P)

- 1) F=0 initialization
- 2) Create header table holds for all items i in tree
- 3) Generate a new item set I with prefix and support of item i
- 4) I – Infrequent item
- 5) Construct I as conditional pattern and FP tree
- 6) Select the infrequent items from the set
- 7) Remove from Tree and finally apply recursive mining



International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2014

VII.CONCLUSION

In this survey is focused on the infrequent weighted item sets, from transactional weighted data sets to address IWI support measure is defined as a weighted frequency of occurrence of an item set in the analyzed data. Occurrence weights derived from the weights associated with items in each transaction and applying a given cost function. Discussed in detail on various frequent item set and infrequent item set mining such as Apriori, FP tree, FP growth, mining infrequent item set, Infrequent weighted item set IWI and Minimal infrequent item set MIWI

REFERENCES

- [1] R. Agrawal, T. Imielinski, and A. Swami, "Mining Association Rules between Sets of Items in Large Datasets," Proc. ACM SIGMOD '93, pp. 207-216, 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.
- [3] K. Sun and F. Bai, "Mining Weighted Association Rules Without Preassigned Weights," IEEE Trans. Knowledge and Data Eng., vol. 20, no. 4, pp. 489-495, Apr. 2008.
- [4] J. Han, J. Pei, and Y. Yin, "Mining Frequent Patterns without Candidate Generation," Proc. ACM SIGMOD Int'l Conf. Management of Data, pp. 1-12, 2000.
- [5] D.J. Haglin and A.M. Manning, "On Minimal Infrequent Itemset Mining," Proc. Int'l Conf. Data Mining (DMIN '07), pp. 141-147, 2007.
- [6] BalazsRacz, "nonordfp: An FP-Growth Variation without Rebuilding the FP-Tree", 2nd Int'l Workshop on Frequent Itemset Mining Implementations FIMI2004
- [7] J. Han, J. Pei, and Y. Yin, "Mining Frequent Patterns without Candidate Generation," Proc. ACM SIGMOD Int'l Conf. Management of Data, pp. 1-12, 2000.
- [8] A. Erwin, R.P. Gopalan, and N.R. Achuthan, "Efficient Mining of High Utility Item sets from Large Data Sets," Proc. 12th Pacific-Asia Conf. Advances in Knowledge Discovery and Data Mining (PAKDD), pp. 554-561, 2008
- [9] R. Agrawal and R. Srikant, "Mining Sequential Patterns," Proc. 11th Int'l Conf. Data Eng., pp. 3-14, Mar. 1995.