AIF Algorithm - An efficient approach to increase the sales of infrequent items.

A.Lavanya
Assistant Professor,
: Dept.of Computer Science and Engineering
Sriram Engineering college
perumalpattu, India
rebica.it@gmail.com

G.Muthukumar
Assistant Professor:
Dept.of Computer Science and Engineering
Sriram Engineering college
perumalpattu, India
gmuthu79i@gmail.com

S.Amudha
Assistant Professor:
Dept. of Computer Science and Engineering
Sriram Engineering college
perumalpattu, India
amudhasaravanan@gmail.com

Received 21 February 2019 Received in revised form 24 February 2019 Accepted 25 February 2019
Available online 03 March 2019

ABSTRACT

Frequent itemset mining is one of the active research area in data mining. There exist a purpose to look after different data and its applications. At the same time, infrequent itemset mining plays an eminent role . Based on the minimum threshold the frequent items are generated using dFIN Algorithm. The infrequent items are extracted from the transaction dataset. The sales of infrequent items are promoted using AIF(Association of infrequent item with frequent itemset) Algorithm. The infrequent items are extracted based on the threshold. The infrequent item is mapped with the largest frequent item. The mapping is based on the least expiry date of infrequent item and the largest support count of frequent items. Through this way the sales of infrequent items gets increased.

Keywords—Frequent itemset, DiffNodeset, Data mining, Algorithm, Market Basket Analysis.

I. INTRODUCTION

Data mining is one of the exploring topic in Computer science.The different data from different domain are extracted based different applications.The data may be of different types namely geographic data,spatial data,scientific data,medical data,games.The data are stored in files,database and repositories.The data are cleaned to remove the noise and is integrated from multiple sources.Then it is transformed to appropriate form.It is also called as data consolidation.The patterns are evaluated under different consideration.Finally knowledge representation is discovered.Hidden data is not readily evident.Many data are of high dimensionality.The data from different dimensions are analaysed.In data mining, there is an active topic called Frequent itemset mining.In general, Frequent itemset mining is known to be the items purchased by the customers frequently.If a customer buys milk and biscuit, then he/she will probably buy bread.The frequent items are generated using different frequent miner algorithm using different structures.Each Algorithm has its own merits and demerits.The term support is defined as the frequency of occurrence of each item in a transaction.

supp(X)= no. of transactions which holds the itemset X / total no. of transactions

Spotting frequent itemset is one of the most important cause faced by the data mining community.Out of all the infrequent items are the items which are below the minimum threshold.When the support is high, less number of frequent itemset will be generated.When the support is low, then large number of frequent itemset is generated.The itemset are mined to discover different patterns according to the given threshold.The behavior of customers can be tracked using frequent items purchased by the customers.

Initially the transaction dataset is given for frequent itemset generation.After scanning the dataset, a tree is constructed composing of 1-itemset.After finding 1-itemset, 2-itemset are discovered.Then frequent itemset are generated till k-itemset.It is done using algorithm used for frequent itemset mining and stitemsetructures proposed for each algorithm.The infrequent items which is lying below the threshold are extracted and weighted according to the count of each item.

In existing system, there are algorithms introduced for mining frequent itemsets.Each algorithm uses appropriate structure for mining the items.In prepost algorithm, a structure called N-list is introduced for storing all information about itemset.Prepost can find frequent itemset without generating candidate itemset by making use of single path property of N-list.In FP-growth algorithm divide and conquer approach is used for mining frequent itemset.In Eclat-g algorithm depth first approach is used for mining frequent items.The intersecting support of two of its subset is used for mining items.In FIN Algorithm, a structure called Nodeset is introduced for mining items.Nodeset requires only preorder of each node which makes it retain half of the memory.

In this paper,we propose an structure called DiffNodeset to mine the items which are frequently purchased.The items are retrieved based on the dFIN Algorithm.The dFIN Algorithm works based on the set enumeration tree and superset equivalence property.It requires only the preorder of each node for the DiffNodeset of each item. The infrequent items are retrieved based on the minimum threshold.The sales efficiency of infrequent items are performed by associating an infrequent item with least expiry date and the largest frequent pattern.It is performed based on the category of each infrequent items.The infrequent item of a category is mapped with the frequent item of large support of the same category.

II. LITERATURE SURVEY

Every itemsets are mined using DiffNodeset structure. dFIN Algorithm is suggested in order to mine the itemsets efficiently(Zhi-Hong Deng 2016)[11].At the outset the PPC tree is built to mine F1 itemset.The database is examined to mine all 1-itemset with support count.The infrequent items are detached based on the minimum support count.Based on the form of DiffNodeset structure 2-itemset are drawn by ancestor descendant relationship.Finally k-itemset are mined using set enumeration tree.All possible pattern can be observed using set enumeration tree.A vertical algorithm called PPV is proposed for fast frequent pattern discovery. PPV acquire Node-lists of each frequent itemset.(Z.H.Deng , Z.H.Wang 2010)[4].Then PPV obtains Node-lists of the candidate patterns of length k and discovers the frequent patterns of length (K+1).

An efficient data structure called nodeset is proposed.Nodeset requires only preorder which consumes half of the memory when compared with N-list and Node list.Based on the Nodeset structure an efficient algorithm called FIN is proposed for mining frequent itemset efficiently(Zhi-Hong Deng,Sheng-Long Lv(2014)[2].FIN adopts promotion which is based on superset equivalence property as pruning strategy.

Prepost+, a high performance algorithm is introduced for mining frequent itemset.It employs N-list to represent itemset and discovers frequent itemset using set-enumeration search tree.Especially it employs an efficient pruning strategy named Children-Parent Equivalence pruning to greatly reduce the search space.(Zhi-Hong Deng , Sheng-Long Lv 2015)[3].This work of Prepost+ is same as that of Prepost.Mining erasable itemset using NC-sets is proposed, which keeps track of complete information used for mining erasable itemsets efficiently.(Zhi Hong Deng,Xiao Ran Xu 2012)[6].The efficiency of MERIT is achieved with three techniques.

An algorithm is introduced for mining frequent itemsets is presented in a stream of transactions within a limited time horizon.(Luigi Troiano, Giacomo Scibelli (2014))[13]. The proposed algorithm makes use of a test window get rid of non-frequent itemsets from a set of candidates.When the the support threshold is high, the test window is smaller. In addition to considering a sharp horizon, a smooth window is considered. Smoothness is ascertained in both qualitative and quantitative terms.The Window Itemset Shift (WIS) as an substitute solution, which retains a memory of flowing candidates within a reduced test window.This work,the problem of mining frequent itemsets in a flow of transactions is within a limited window. In addition, WIS does not require a pass through the dataset to compute the support.

Processing incremental databases in the itemset mining is important because a huge amount of data has been assembled continuosly in a variety of application fields and users want to obtain mining outcome from incremental data in efficient way. One of the major problems in incremental itemset mining is that the mining results is far reaching according to threshold settings and data volumes. Moreover, it is hard to analyze information. Furthermore, not all of the mining results become significant information. In this work, to solve these difficulties, an algorithm is proposed for mining weighted maximal frequent itemset from incremental databases.(Unil Yun, Gangin Lee (2016))[14].

Two novel approaches are proposed to drive the IWI mining process. Two algorithms are proposed that perform IWI and Minimal IWI mining effectively, handled by the proposed measures, are presented.(Luca Cagliero and Paolo Garza (2014))[7].Given a weighted transaction data set and a maximum IWI-support threshold the Infrequent Weighted Itemset Miner algorithm finds all IWIs whose IWI-support satisfies minimum support threshold.

The infrequent weighted itemset are item sets whose frequency of existence in the analyzed data is less than or equal to a maximum threshold. Two algorithms are inspected to find rare itemset, that are infrequent weighted itemset (IWI) and Minimal Infrequent Weighted Itemset (MIWI) and is based on the frequent pattern-growth paradigm. IWI Miner is a FP-growth-like mining algorithm that performs projection-based item set mining.(Nandhini S,Yogesh M and Gunasekaran S. (2015))[9]. FP-growth mining steps are FP-tree creation then Recursive item set mining from the FP tree index and IWI Miner finds infrequent weighted item sets instead of frequent (unweighted) ones.

III. PROPOSED SYSTEM

The proposed framework for frequent itemset generation is based on dFIN Algorithm.It uses DiffNodeset structure to mine the frequent itemset.Initially the transaction database is scanned to construct the PPC(Preorder postorder code) tree.It is constructed based on the minimum threshold.The 1-itemset are sorted in support descending order.Then the 2-itemset are constructed based on the DiffNodeset structure.Here the non-ancestor nodes are taken and calculated.Then the k-itemset are constructed using pattern tree.It employs two techniques namely set enumeration tree and superset equivalence property.

Hence all the possible combination of itemset can be described using set enumeration tree. The infrequent items are extracted which are lying below the minimum threshold. The infrequent item with least expiry date is taken and mapped with the frequent item of large support.It is done using AIF(Association of infrequent item with frequent item).Through this way the sales efficiency of infrequent items gets increased.The frequent itemset generation and infrequent items promotion is proposed as the design and is shown in Figure 3.1

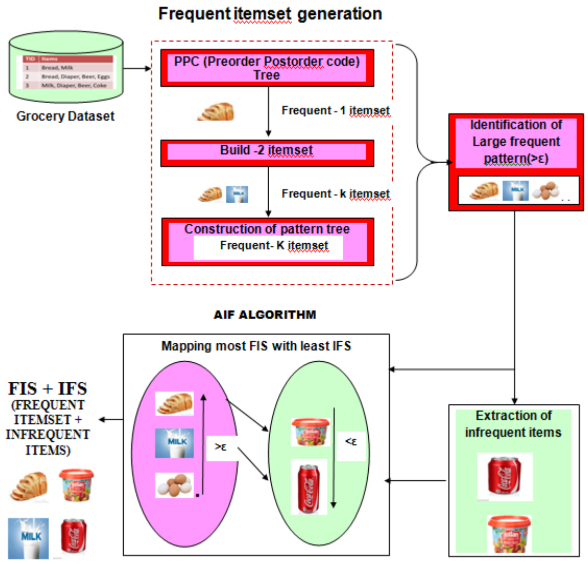

Figure3.1.Proposed Architecture

A. *ppc tree construction:*

ppc tree (preorder postorder codes)

In the Construction of PPC tree, the input transaction dataset is scanned to find 1-itemset.All the F_1-items are retrieved based on the given minimum support threshold(**ε**). The F_1-items are sorted in support descending order. All the infrequent items are deleted and the sorted frequent items are placed in the PPC tree. The PPC tree is scanned to generate the preorder and postorder codes by the preorder traversal.

B. *Build_2_itemset:*

In Build_2_itemset, the nodeset of two itemset are compared. The nodeset comprises of preorder code, postorder code, count of each item. The DiffNodeset of 2-itemset i_1i_2, denoted as DiffNodesetsi_1i_2. DiffNodesetsi_1i_2= {(x.pre-order, x.count)|x ∈ Nodesetsi_1 ∧¬ (∃y ∈ Nodesetsi_2, the node respect to y is an ancestor of the node co rresponding to x)}.

where Nodesets$i_1$and Nodesetsi_2 are the Nodesets of items i_1 and i_2 respectively.

In addition, the elements in DiffNodesetsi_1i_2 are sorted in pre-order ascendant order. The non-ancestor nodes are taken as DiffNodeset. Therefore the support of 2-itemset is calculated as follows:

The support of i_1i_2, support(i_1i_2), is equal to

$$\text{support}(i_1) - \sum (E \in DN_{12})\ E.\text{count} \text{------ } 3.1$$

Equation 3.1 shows the support of 2-itemset generation. The DiffNodeset of 2-itemset is calculated and subtracted from the support of first item.

C. *construction of pattern tree:*

In construction of pattern tree, the k- itemset (k≥3) are generated. It employs set enumeration tree and superset equivalence property. It generates k-itemset (k≥3) extended from frequent 2-itemset.The support is also calculated for all itemset. superset equivalence property is employed to prune the search space. all the possible pattern of the frequent items can be observed using set enu meration tree.

D. *AIF algorithm :*

With Frequent itemset, it is possible to identify infrequent items that have support less than threshold. By associating an infrequent item with a frequent itemset, the proposed work improves the sales of infrequent item. The association is based on the expiry date of infrequent itemset and support count of frequent itemset.

Each item belong to particular category. Based on the category the infrequent items are selected and matched with the frequent itemset of same category.The infrequent item which has got the least expiry date is selected and associated with the maximum frequent item of the category.If that association is not matched, the next maximum frequent item is selected.By doing this the sales of infrequent items is increased.

INPUT : Infrequent items, Database with expiry date D, minimum support threshold(ε)
OUTPUT : Association of infrequent item with Frequent itemset.

```
Start
Scan database with items and expiry date
Set ε = minimum  threshold value
Infrequent items are retrieved
For all infrequent items in database D
do
        If expiry date of infrequent item has least value then
         Map infrequent item with Large frequent pattern(K)
                Else
                   Map infrequent item with Large frequent
pattern(k-1)
End if
End for
End
```

The above algorithm is made run for every infrequent item with least expiry date. If the item is not sold then it is associated with other large pattern. Hence the sales rate of infrequent items will be increased.

SNO	ITEMNAME	CATEGORY	FROMSTOCK	EXPIRYDATE	QUANTITY	SOLD
1	MILK	1	2-Nov-16	3-Jun-17	18	5
2	BREAD	2	1-Dec-16	4-Jun-17	20	13
3	BISCUIT	2	31-Dec-16	1-Jun-17	10	8
4	CORNFLAKES	3	6-Feb-17	14-Jul-17	10	6

Table 3.1 Items with expiry date

The above table 3.1 shows the items with expiry date. It has the list of items along with its category, expiry date, quantity and number of items which is been sold.

The items are sorted according to the given threshold.Then the infrequent item which has got the least expiry date is taken and mapped with the frequent item which has got the maximum count.The mapping takes place according to the category.If the infrequent item is less when compared to frequent items, then the infrequent items will be mapped according to its category. If the infrequent items are more when compared to frequent items then the infrequent items are mapped to the existing frequent items and the remaining infrequent items are mapped within themselves.

The main objective of this work is to promote the sales of infrequent items with some offer. Hence the sales of infrequent items is increased and sold with offer.

IV. EXPERIMENTAL STUDY

For the frequent itemset generation we collected dataset fromhttp://fimi.uc.ac.be/srcandhttp://www.adrem.ua.ac.be/goethals /software respectively[11].It contains the dataset like chess, pumsb ,kosarak, mushroom and T10I4D100K .In order to evaluate the performance of dFIN it is checked out with all possible datasets.The outcome of dFIN is observed with the dataset. The infrequent items are extracted from the dataset and the sales efficiency of the infrequent items are promoted by associating the infrequent item which has got least expiry date with the greatest support count.dFIN Algorithm works best when compared to existing leading algorithms.It consumes less memory and running time.Hence Frequent items are generated using dFIN algorithm and the sales efficiency of infrequent items are increased using AIF Algorithm.

V. CONCLUSION

The frequent itemsets are mined efficiently using DiffNodeset structure.Based on the structure of DiffNodeset, dFIN algorithm is presented for generating frequent itemset efficiently. dFIN observes frequent itemset using set enumeration tree and superset equivalence property. The running time and memory consumption is comparatively reduced with existing leading algorithms. Based on the minimum support threshold value, the infrequent items are pruned and frequent items are generated.With frequent itemset it is possible to extract infrequent items that have support less than threshold value. By associating an infrequent item with a frequent itemset, the proposed work improves the sales of infrequent items. The association is based on the expiry date of infrequent itemset and support count of frequent itemset.

REFERENCES

[1] Deng Z.H.(2014),'Top-Rank-K frequent patterns by using Node-lists Expert syst.Appl,vol.41,No.4,pp.1763- 1768.
[2] Deng Z.H and Lv S.L.(2014), 'Fast mining frequent itemsets using Nodesets', Expert syst.Appl, vol.41,No.10,pp.4505-4512.
[3] Deng Z.H and Lv S.L. (2015), 'Prepost+:an efficient N-Lists based algorithm for mining frequent itemsets via children-parent equivalence pruning',Expert syst.Appl, vol.42,No.13,pp.5424-5432.
[4] Deng Z.H. and Wang Z.H.(2010), 'A new fast vertical method for mining frequentitemsets',Int.J.Comput.Intell.Syst,vol.3,No.6, pp.7 33-744.
[5] Deng Z.H. ,Wang Z.H. and Jiang J.J.(2012), 'A new algorithm for fast mining frequent itemsets using N-Lists' ,Sci.China Inf.Sci,vol.55,No.9,pp.2008-2030.
[6] Deng Z.H and Xu X.R(2012), 'Fast mining erasable itemsets using NC-sets',Expert syst.Appl,vol.39,No.39,No.4,pp.4453-4463.
[7] Luca Cagliero and Paolo Garza (2014), 'Infrequent weighted itemset mining using frequent pattern growth' ,Knowledge and data engineering, vol.26,No.4,pp.903-915.
[8] Lee A.J.T ,Wang C.S ,Weng W.Y.Chen Y.A and H.W. (2008), 'An Efficient Algorithm forming closed intrer-transaction itemsets' ,Data Knowl.Eng,vol.66,No.1,pp.68-91.
[9] Nandhini S,Yogesh M and Gunasekaran S. (2015), 'Survey on infrequent itemset mining using frequent pattern growth' ,Journal of scientific and Engineering Research,vol.6,No.4,pp.23-25.
[10] W.Lin J.C, P.Heng T. and C.Lin T, 'A CMFFP tree algorithm to mine complete multiple fuzzy frequent itemsets',Appl.Soft Comput,vol.28,pp.431-439.
[11] Zhi-Hong Deng (2016), 'DiffNodesets:An Efficient structure for fast mining frequent itemsets',Appl.Soft.Comput,vol.41,pp.214-223.
[12] Zaki M.J and Gouda k.(2003), 'Fast vertical mining using diffsets',SIGKDD,pp.326-335.
[13] Luigi Troiano, Giacomo Scibelli(2014),'Mining frequent itemsets in data streams within a time horizon',Data Knowl.Eng,vol.42,pp.21-37.
[14] Unil Yun,Gangin Lee(2016), 'Incremental mining of weighted maximal frequent itemset from dynamic databases',Expert Syst with Applications,vol.54 pp.304-327.
[15] www.andreson.ucla.edu/faculty/jasonfrand/teacher /technologies/palace/datamining.htm.