Volume 4, Issue 12, December 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Cost Efficiency of Data Mining Within One Pass

Ms. Shinde Rupali R, Prof. Maral Vikas B

Dept. of Computer Engineering, K.J. College of Engineering and Management Research, Pune, Maharashtra, India

Abstract : Data mining is nothing but the extracting the data from the data warehouse. But the basic of data mining is to extract the data pattern which are weighted frequent pattern in the data warehouse. Because this type of extraction is very useful for business analysis. And weighted frequent pattern mining is tremendous practical than the frequent pattern mining. In the frequent pattern mining only the support of item or itemset is considered. But in weighted frequent pattern mining specific significance i.e price , useness will be considered for the weight calculation. So the combination of weight and fre quent pattern mining is very important . We have advantage of Weighted frequent pattern mining (WFM) is it used for the stream data mining which is static .There are various types of algorithms which are used for the weight frequent pattern mining e.g FUN,FUN2,UFEP,Apriori based algo.,FPtree algorithms but which are not adoptable for the for the simultaneously upadated database of transaction database or it required more than two scan for mining the updated database. So we are going to do the research work on the updated data of transaction database(TDB). Now tree data structure is used for the arranging the data items of TDB. We have two algorithms for data mining one UWFPMTWA(Updated Weighted Frequent Patter mining tree with Weight ascending order) And UWFPMTFD(Updated Weighted Frequent Pattern mining tree with Frequency descending order) .This algorithms required less memory space than the others because tree is having the prefix sharing of path for the same items which occurred in different transactions. In that research work we have to create tree of updated as well as interactive database.

Keywords : Data mining ,Weighted frequent pattern mining,Updated TDB, Mining with tree structure.

I. INTRODUCTION

In the introduction section we have to see the different types of algorithms. But have some drawbacks regarding that so in this paper we are developing two algorithms i.e. UWFPMTWA (Updated Weighted Frequent Pat-tern mining tree with Weight ascending order) And UWFPMTFD(Updated Weighted Frequent Pattern mining tree with Frequency descending order). WFP mining becomes an important research concept in data mining and knowledge discovery. However, previous algorithms doesn't work for increased WFP mining and also for continuous data that is stream mining reason of that is whole data are based on a stable database and needed more than one database scans. In this research paper , we present two novel tree structures UWFPMTWA (Updated WFP tree based on weight ascending order) and UWFPMT FD (Updated WFP tree based on frequency descending order), and two new algorithms UWFPMTWA and UWFPMTFD for Updated WFP mining using a single Transaction database scan. They are effective for current any transaction DB and interactive mining to utilize the latest tree structure and to use the mining results of latest mining operation when a database is updated or a minimum support threshold is changed. Let I =(item1 , item2 , ... , item m) be a set of items and D be a transaction database (Tran1, Tran2,….Tran n) subset of the set X =(x1 , x 2, ... , x k), where X belong I and k [1, m]. However, an itemset is called k itemset when it contains k distinct items. For example, ab is a 2 item and abde is a 4 item pattern in A weight of an item is a positive integer number assigned to respect the importance of the item in the transaction database. The weight of a pattern, P(item1 , item2 , ... , item k) is given as follows:[1]

$$\text{Weight(pat)}=\sum_{q=1}^{\text{length(pat)}}/\text{length(pat)}.$$

For example, Weight(ad) = (0.6+0.35)/2 = 0.475 in the example database. A weighted frequency of a pattern is defined as the final value of multiplying the patterns frequency with the weight of the pattern. So the weighted frequency of a pattern (Pat) is given as : WS (Pat)=Weight(pat)*support(pat) For example, WS(ad)=0.475 * 4 = 1.9.

II. LITERATURE SURVEY:SURVEY OF SIMILAR SYSTEMS ALONG WITH PROS AND CONS

We have to see the history of the frequent pattern data mining Fast algorithms for mining association rules for the association mining rules the problem is to generate all the association rules that have support and confidence greater than user defined min support and confidence .For the association rule mining AIS and SETM Algorithms are developed. Apriori hybrid algorithm is advance for the scale up property fleexible and combination of Apriori and Apriori Tid algorithms.[4] Suppose Transaction T contains X , X is sub-set of T the association rule XY where X is subset of T and Y is subset of T. And X intersection Y NULL. XY this association put in the transaction DB with Confidence C if c per of transaction contains the same association.so in the same algorithms for each item we have to and the association rules with other items.

2.1 Sequence pattern mining

Incremental mining for sequential pattern mining in the sequential pattern mining the patterns are analyzed on the basis of time constraints of the items. So Generalized sequential pattern(GSP) for the sequential pattern mining .Which are Apriori based Algo. One more drawback of GSP is to maintain the sequence of pattern for long time or for the long time the data was may becomes outdated. So another efficient algorithm i.e. ISE(Incremental sequential extraction) is invented .Suppose S is set of item are ordered ac-cording to time (instances(s1,s2,s3,s4,..sn) if the customer punches the items (1,2,3,4,5) in order to((1),(1.2),(3)(4,5)).[5]

2.2 Canonical Tree

Frequent pattern mining has been a focused term in data mining. An efficient and flexible algorithms for high support itemset mining in transaction databases such as continuous data pattern mining, structured pattern mining, correlated mining, associative classification, and frequent pattern based clustering, as well as their broad applications. Canonical Tree Previously many algorithms are developed i.e. FUN,FUN2,UFEP for the updated TDB mining but all are the Apriori based so for the candidate item generation generate and test are required for each level. And next algorithms are LINE,FELINE are well suited for the interactive mining i.e. build once mine many. Means TDB will static but minimum support is changed. But in the Canonical tree all the candidate items are analyzed on the basis of frequency test.[2]. Apriori based algorithm is used but it is not suitable for the FP Tree(Frequent pattern Tree).[2]. such as create and check of all candidates and multiple scanning a large amount of the original transaction database. And apriori based algorithm is used association rules for mining, the mining ,But association rules is time consuming. In that case only the frequency of the item is considered .Frequency is nothing but the support of item i.e. how much time that item is occurred in various transaction is nothing but the frequency. The support per frequency of a pattern is the how many times that pattern occurred in transaction database. The problem of highly support pattern mining is to find the all set of patterns which will full fill a minimum support in the transaction database. Minimum threshold support is nothing but the user specified non negative value.

2.3 Weighted frequent incremental mining

WFIM is(Weighted Frequent Itemset Mining)[9] method to use a pattern addition algorithm. WFIM only work on the downward closure property while maintaining algorithm facility. Patterns created by WFIM have weak frequency and weight affinity patterns. WFIM uses a weight's upper limit and lower limit to manage the various of patterns. However, WFIM doesn't give the facility to drop out patterns that include items with different frequency and weight levels. It would be good option if the less affinity patterns could be deleted from list, resulting in fewer patterns after mining. We have to use another algorithm for the weighted frequent pattern mining i.e. Weighted Interesting Pattern mining(WIP).[7] We de ne the aspect of a weighted hyper clique pattern that uses a new count , called weight assurance, to concentrate on weight and prevent the creation of patterns with every step different weight levels In the WIP above drawbacks are overcome. But above the algorithms are separately developed for the Frequent pattern mining or Weighted pattern mining. Or Weighted frequent pattern mining. Which may use Association rules, and the candidate items or super set of item which having the support which greater than minimum support.

2.4 Peaon count Tree for multimedia mining

Various algorithms for the multimedia data mining, The Data SURG group at NDSU has a long term interested in data mining remotely sensed imagery (RSI) for agricultural, forestry and other knowledge discovery and studing various real life example. A spatial data mining, the Peano count tree, was invented that give an efficient , lossless, data mining and show different types of data of that application. This data representation has vital for the mining of more and very large data stock, with time instances of RSI and multimedia data. The P tree technology assures an exactly way to keep and mine multi-media of any format, together with Geographical images as a data formats. Our proposed structure for the data mining is to use tree for arranging items according to Weight Ascending order And Frequency Descending order. This type of techniques can be mined for the sequential pattern , structure pattern multimedia data pattern.

In UWFPMTWA gets advantage in candidate pattern generation by producing the largest weighted item in the last child of tree of UWFPMTWA. UWFPMTFD give the guarantee that any candidate item which weight is less than min threshold cannot appear before candidate items in any stream of UWFPMTFD tree and thus speeds up the reduced tree of any particular item and tree of all candidate item of that particular item generation time during mining operation. UWFPMTFD also gain the more compressed updated tree to reduce memory space. To our knowledge, this is the research work to perform single pass incremental mining for weighted frequent patterns .The current approach support the Downward closure property .

III. IMPLEMENTATION DETAILS

Implementation of Updated Weighted Frequent Pattern mining tree we want following inputs:

- Transaction database (TDB). - Weight of Item
- Frequency of item.
- Header table.

3.1 Weighted Frequent pattern tree based on WA order.

In the updated frequent pattern mining we have to arrange the weight of the item in the ascending order because

of that we can get the biggest weight at the bottom of the tree and bottom up mining is easy to mine the item .As per above requirement first of all we want transaction table which is having all the transactions

	tid	transactions	
	TRN00001	Scandisk Pendrive,HCl Laptop,I-ball Headphone,Sony Speakers,Len...	73 b...
	TRN00002	Kasper Key Antivirus,HP Desktop,Samsung Galaxy Y,Lenovo Laptop	62 b...
	TRN00003	HCl Laptop,Samsung Galaxy Y	27 b...
	TRN00004	I-ball Headphone,Scandisk Pendrive	34 b...
	TRN00005	HCL Desktop Computer,Dell Desktop Computer	42 b...
	TRN00006	HCl Laptop,Scandisk Pendrive	28 b...
	TRN00007	HCL Desktop Computer,I-ball Headphone,Scandisk Pendrive	55 b...
	TRN00008	Dell Desktop Computer	21 b...
	TRN00009	Lenovo Laptop,Quick Heal Antivirus,Samsung Galaxy Y	51 b...
	TRN00010	HCl Laptop,I-ball Headphone	27 b...
	TRN00011	HCL Desktop Computer,I-ball Headphone,Sony Speakers	51 b...
	TRN00012	HP Desktop,HCl Laptop	21 b...
	TRN00013	HCL Desktop Computer,I-ball Headphone,HP Desktop	48 b...
	TRN00014	Kasper Key Antivirus,HP Desktop,Samsung Galaxy Y,Scandisk Pendrive	66 b...
*		(NULL)	0 Kb...

Figure 1: Transaction Database

After that we want weight table as per I have mention above weight is nothing but the price , useness , in the web analysis number of click on any link etc.In our paper we have to calculate the weight on the basis of price in the retail database. For calculation the weight of the each item we have to divide the price 10000. For the constant value.

	item_id	item_name	item_price	item_weight	item_image	
	ITM00001	Kasper Key Antivirus	1300	0.13	(Binary/Image)	8 Kb...
	ITM00002	HCL Desktop Computer	15000	0.15	(Binary/Image)	6 Kb...
	ITM00003	I-ball Headphone	150	0.015	(Binary/Image)	5 Kb...
	ITM00004	HCl Laptop	30000	0.3	(Binary/Image)	7 Kb...
	ITM00005	Dell Desktop Computer	19000	0.19	(Binary/Image)	6 Kb...
	ITM00006	HP Desktop	21000	0.21	(Binary/Image)	6 Kb...
	ITM00007	Lenovo Laptop	35000	0.35	(Binary/Image)	7 Kb...
	ITM00008	Samsung Galaxy Y	21000	0.21	(Binary/Image)	6 Kb...
	ITM00009	Scandisk Pendrive	500	0.05	(Binary/Image)	6 Kb...
	ITM00010	Quick Heal Antivirus	1100	0.11	(Binary/Image)	8 Kb...
	ITM00011	Sony Speakers	4500	0.45	(Binary/Image)	4 Kb...
*		(NULL)	(NULL)	(NULL)	(NULL)	0 Kb...

Figure 2: Weight table of item

In this paper we have to create at the end of transaction .Tree creation after T1 transaction on the basis of Weight Ascending order is : for that purpose we want header table in which we can sort items in the weight ascending order by using bubble sort algorithms.so the header table after T1 transaction.The header table will generated at the end of each transaction.

ITEM	FREQUENCY
I-ball Headphone	6
Scandisk pendrive	5
HP Desktop	5
HCL Laptop	5
HCL Desktop computer	4
Samsung Galaxy	4
Dell Desktop Computer	3
Lenova Laptop	3
Kasper Key Antivirus	2
Sony Speakers	2
Quick Heal Antivirus	1

Fig:3 Frequency table


```
Header table updated

curr_trans : Scandisk Pendrive,HCl Laptop,I-ball Headphone,Sony Speakers,Lenovo Laptop

-----------------------------------------------------------------
|  Item                          |  Weight     |  Frequency   |
-----------------------------------------------------------------
|  I-ball Headphone              |  0.015      |  1           |
|  Scandisk Pendrive             |  0.05       |  1           |
|  HCl Laptop                    |  0.3        |  1           |
|  Lenovo Laptop                 |  0.35       |  1           |
|  Sony Speakers                 |  0.45       |  1           |
-----------------------------------------------------------------
```

Figure 4: Header table of T1 transaction

T1=Scandisk Pen drive , HCL Laptop ,I-ball Headphone ,Sony Speakers ,Lenovo Laptop After arranging Ascending order weights I-ball Headphone ,Scandisk Pen drive , HCL Laptop , Lenovo Laptop , Sony Speakers .

```
0 ----0.0---- null
0 ----0.015---- I-ball Headphone
0 ----0.05---- Scandisk Pendrive
0 ----0.11---- Quick Heal Antivirus
0 ----0.13---- Kasper Key Antivirus
0 ----0.15---- HCL Desktop Computer
0 ----0.19---- Dell Desktop Computer
0 ----0.21---- HP Desktop
0 ----0.3---- HCl Laptop
0 ----0.35---- Lenovo Laptop
0 ----0.45---- Sony Speakers
0 ----0.0---- null
0 ----0.015---- I-ball Headphone
0 ----0.05---- Scandisk Pendrive
0 ----0.11---- Quick Heal Antivirus
0 ----0.13---- Kasper Key Antivirus
0 ----0.15---- HCL Desktop Computer
0 ----0.19---- Dell Desktop Computer
0 ----0.21---- HP Desktop
0 ----0.3---- HCl Laptop
0 ----0.35---- Lenovo Laptop
0 ----0.0---- null
0 ----0.0---- null
0 ----0.015---- I-ball Headphone
0 ----0.05---- Scandisk Pendrive
0 ----0.11---- Quick Heal Antivirus
0 ----0.13---- Kasper Key Antivirus
0 ----0.15---- HCL Desktop Computer
0 ----0.19---- Dell Desktop Computer
0 ----0.21---- HP Desktop
0 ----0.3---- HCl Laptop
```

Figure 6: Tree creation after T3 transaction.

In the tree creation time first of all the tree is null when the header table of the first transaction come according to weight ascending order after null element first item will be inserted , and single path will be generated for the next transaction if the first item will be same then that item will common to both path that's called prefix sharing that's why the tree will be compressed and memory space will be reduced.

3.2 Weighted Frequent pattern tree on the basic of FD order

Weighted frequent patter mining on the basis of frequency descending order .In this algorithm we have to arrange items in the frequency descending order .In the frequency descending order we have to arrange the frequency in descending order so we can get highest frequency at the top of the tree.So the top up mining easy for the pattern recognition . Also the tree of the FD order is compressed as compare to the WA algorithm. And Non candidate item will not come in between candidate item.In the following diagram we have frequency table after all transaction of whole TDB.

3.3 Mining strategy of UWFPMTWA.

In the UWFPMTWA Algorithms we have new concept for achieving the Down word closure property that if the any item is infrequent then all the superset of that item will be infrequent i.e. GmaxW(Global maximum weight) of the item of TDB.in our example a=0.6 is GmaxW .By using GmaxW we have to get items which are not weighted but frequent. Local maximum weight, nothing but LMAXW, is required when we will perform the mining operation for a particular item. For the mining we have to create the Prefix tree and conditional tree, prefix tree is nothing but the all previous branches of particular item .Conditional tree is minimal of prefix tree which delete the non-candidate items .Suppose the minimum threshold = 0.5.And that infrequent items are those whose GMAXW * Frequency < Threshold value .

3.4 Mining strategy of UWFPMTFD.

At now we are mini the data according to frequency descending order. One advantage of the UWFPMT FD more path sharing for the item arrangement so less memory space are required. UWFPMT FD is having the same procedure for generating prefix and conditional trees in mining . Normallyit is sorted according to the frequency descending order, LmaxW could be anywhere for a particular item.The UWFPMT FD is top to bottom mining now we start the pattern growth mining operation from the top-most item.

IV. SYSTEM ARCHITECTURE

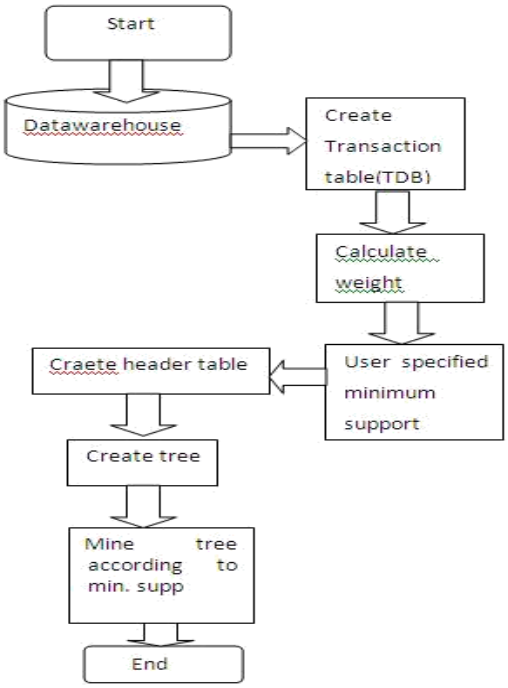

Figure 7: Basic Architecture

In the above architecture normally for the business analysis we have the data warehouse which is having very large dataset from that data set we want to recognize the frequent and weighted pattern. That's called pattern mining . In our paper we have transaction DataBase (TDB) as a data warehouse .From that TDB we have to generate tree according to weight ascending order and frequency descending order . Then take the minimum threshold from user for the mining then according to algorithms we have to generate the tree and calculate candidate pattern form the prefix tree . find the weighted frequent pattern mining.

V. ALGORITHMS

1.The tree generation algorithms of UWFPMTWA
TDB,Weight tableof each item,minimum threshold
Output Weighted frequent pattern tree according to weight ascending order.
1)Begin
2)Create the overall header table TDB to keep the items according to the weight ascending order
3)let *GmaxW* be the maximum weight among all the items. 4)Initially craete null root R
UWFPMTWA
5)Check the transaction(Tq) from TDB sort item according to weight ascending order.
6)*if* database is original or db+ then 8)call insert(Tq,R)
9)*else* if Database is db- then
10)all Delete(Tq,R) 11)*else*
12) call modify(Tq,R)
13)*end if*
14)*if* (Tq be your last transaction or any db+,db-,db mod) then 15)*GmaxW*=Weight of the bottom most item in Htab.
16)Input from user and
17)*if* (wetfreq=frequence (a)*GmaxW>) then 18)call test-candidate (a,frequency(a)) 19)create pre x tree.
20)end if
21)CHeack if next transaction.
22)*end*

2. Algorithm for mining
1)Begin
2)*For* each item B of Htab//Conditional tree and its header table contains.
3)*if* (Frequency B)*LmaxW< .
4)Delete B from Htab,Tree
5)*end if*
6)*end for*

7)Let CT be the conditional tree of A //created from T
8)Let HTab be the header table of conditional tree CT
9)For each item B in Htab
10) *Call* testcandiatae(AB,freqyency(AB))
11)Create prefix tree(PT(AB))
12)Call mining(PT(AB),HP(AB),AB,LmaxW)
13)*end for*
14)end

VI. EXPERIMENTAL RESULT AND ANALYSIS

6.1 Analysis environment and data set for the result

Here in this research paper just apply the algorithm on the following data set and compare with previous exiting system and plot the graph according to result. To evaluate our proposed the tree structure we have to use sparse and dense data set .Our algorithm is implemented in JAVA , run on windows 7 operating system , With the Intel (R) Core 2 duo 2.20 Ghz. And 3 GB installed RAM.And 80GB of Hard disc.

6.2 DATA SET:

In our paper we have to use two standard dataset chess and mushroom in the following table I have mention all the characteristics of both the dataset.If R>10% then dataset is dense so our both dataset are sparce.

Data set	Size(MB)	No. of transaction	NO. of distinct item D	Max. transaction length	Min. transactio n length	Avg. trans. length	Dense sparce char. Ration R=(A/D)*100
mushroom	0.56	8124	119	23	23	23	19.327
	0.34	3196	75	37	37	37	49.33

Figure 8: Data set

6.3 Graph

The following graph is shows that the analysis of WA and FD regarding threshold and time for mining. The following graph is shows that the analysis of WA and FD and WFIM algorithm regarding threshold and time for mining. According to following graph WA required more time than FD. Following analysis performed on chess data set.

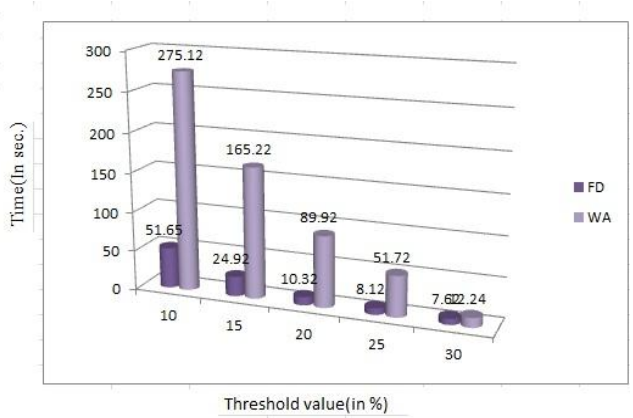

Figure 9: Analysis of WA and FD with respect to treshold and timing.

The following graph is shows that the analysis of WA and FD algorithm regarding number of transaction and time for mining.As shown in the following graph WFIM(Weighted Frequent Incremented Mining) Required more time because it needs two times to scan the up

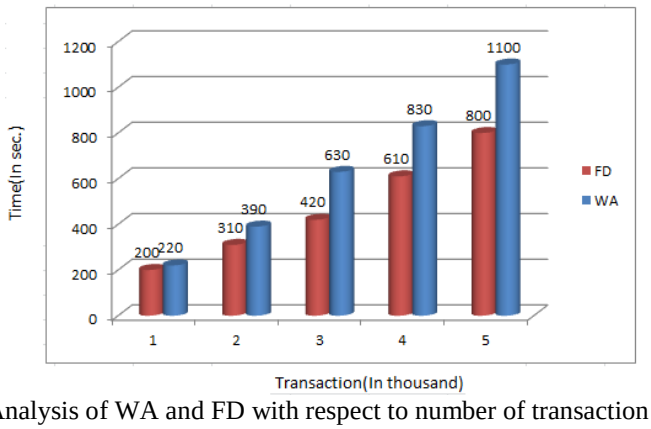

Figure 10: Analysis of WA and FD with respect to number of transaction and timing

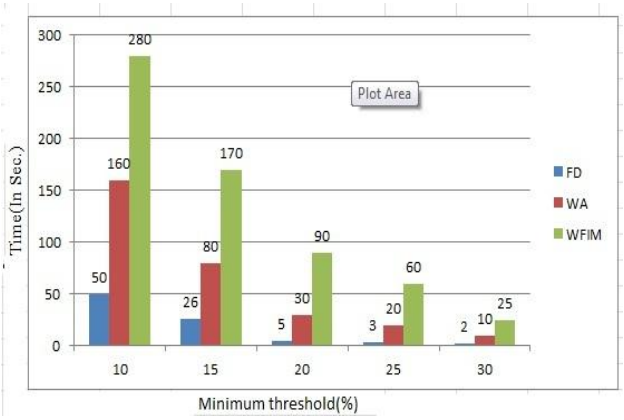

Figure 11: Analysis of WA and WFIM, FD with respect to threshold and timing.

- In the following graph we have to draw time against number of transaction regarding to WA and FD. According to graph FD needs less time than WA. When the transaction increases.

VII. CONCLUSION

Hence the above algorithms are not Apriori based so no need to generate And Test candidate at each level. It efficient because association rules are not used for pattern association. In the tree generation weight as well as frequency is going to be considered.so the infrequent items are not involved in the tree. Tree structure is fast and less space complexity also mining is very fast as compare to the association rule based algorithms. Because in the tree no item occurred in disturbed order .also by using GmaxW we can mini infrequent but weighted items from TDB. The above are used for structured mining ,sequential mining.

REFERENCES

[1] Young-Koo Lee,Byeong-Soo Jeong. "Single pass incremental and interac-tive minig for weighted frequen pattern ."

[2] Leung, C. K.-S., Khan, Q. I., Li, Z., Hoque, T. " CanTree: A canonical-order tree for incremental frequent-pattern mining. "

[3] Dong, J., Han, M." Fast algorithms for frequent itemset mining using FP-Trees".

[4] Agrawal, R., Srikant, R. "Fast algorithms for mining association rules in large databases."

[5] Incremental mining of sequential patterns in large databases"Florent Masseglia a,*, Pascal Poncelet b, Maguelonne Teisseire ca

[6] " INVESTIGATING SIGNIFICANT CHANGES IN USERS INTER-EST ON WEB TRAVERSAL PATTERNS ".

[7] MULTIMEDIA DATA MINING USING P-TREES WILLIAM PER-RIZO, WILLIAM JOCKHECK, AMAL PERERA, DONGMEI REN, WEIHUA WU, YI ZHANG

[8] WIP: mining Weighted Interesting Patterns with a strong weight and/or support a nity Unil Yun and John J. Leggett

[9] WFIM: Weighted Frequent Itemset Mining with a weight range and a minimum weight Unil Yun and John J. Leggett

[10] An E cient Approach for Maintaining Association Rules based on Ad-justing FP-tree Structure Jia-Ling Koh and Shui-Feng Shieh

[11] Chang, L., Wang, T., Yang, D., Luan, H., Tang, S."Efcient algo-rithms for incremental maintenance of closed sequential patterns in large databases. Dataand Knowledge Engineering"

[12] Cheung, W., Zaane, O. R." Incremental mining of frequent patterns without candidate generation or support constraint"

[13] " data mining".

