ISSN 22773061

Improving Efficiency of META Algorithm Using Record Reduction

Shweta[1], Dr. Kanwal Garg[2]

[1] DCSA, KUK, Kurukshetra

shwetabidhan@gmail.com

[2]Assistant professor, DCSA, KUK, Kurukshetra

gargkanval@gmail.com

Abstract— Erasable Itemset Mining is the key approach of data mining in production planning. The erasable itemset mining is the process of finding erasable itemsets that satisfy the constraint i.e. user defined threshold. Efficient algorithm to mine erasable itemsets is extremely important in data mining. Since the META Algorithm was proposed to generate the erasable itemsets. In last few years there have been several methods to improve its performance. But they do not consider the time constraint. If database is large META takes too much time to scan the database. In this paper, Author purposed an Improved META (I-META) algorithm which reduces the scanning time by reduction of production records. It also reduces the redundant generation of sub-items during trimming the candidate itemsets, which can find directly the set of erasable itemsets and removing candidate having a subset that is not erasable.

Keywords — META algorithm, data mining, frequent pattern mining, Apriori algorithm.

Academic discipline— Engineering

Subject Classification—Computer Science

Type—Experimental

Council for Innovative Research

Peer Review Research Publishing System

Journal: INTERNATIONAL JOURNAL OF COMPUTERS & TECHNOLOGY

Vol 8, No 1

editor@cirworld.com

www.cirworld.com, member.cirworld.com

ISSN 22773061

INTRODUCTION

Data mining is emerging field in database system and new database applications [10]. It is the process of finding useful information in large database and finding new relationship among datasets. Frequent pattern mining (FPM) is the important data mining concept which discovers frequent itemset in database [1]. Frequent patterns are the patterns that appear in database frequently. Frequent pattern mining concept worked in many fields such as association rule mining, pattern based classification, clustering, finding correlated item, erasable itemset mining etc [12].

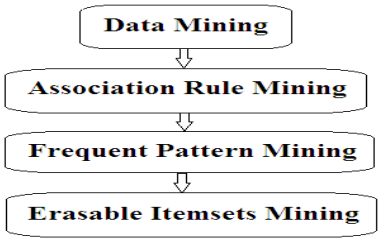

Figure 1 Erasable Itemsets Mining

In this paper author mainly concentrate on erasable itemsets mining problem. Erasable itemsets mining is helpful in production planning in any manufacturing industry. In 2009, G.D. Fang and Z. Deng [2] have introduced the Erasable itemsets mining problem [2]. Any industry manufacturing products (P1, P2, P3…...Pm) which constitutes of some components are known as items (I= i1, i2, i3, i4……in). Sometimes, due to financial crises a manufacturing industry may not be able to purchase all these items. Thus, finding the components which can be erased and without which the loss in profit is controllable is known as erasable itemset mining. And these components are known as erasable itemsets [11]. The original motivation for finding erasable itemset has been raised from the need to control the loss in profit due to absence of some component. Erasable itemsets is helpful for manufacturer to decide how to purchase raw material or help to select which components can be rejected used for manufacturing products in case of some financial problem. META is the candidate generation approach. It generate candidate (k+1)-itemsets based on erasable k-itemsets. In this paper, author proposed a method to improve the efficiency of existing META algorithm for erasable itemsets mining [12].

The organization of the paper is as follow: Section 2 gives literature review. Section 3 defines the problem of mining erasable itemsets. Section 4 give the basic conception related to erasable itemsets mining. Section 5 will give the description and limitation of META algorithm for mining erasable itemsets. Section 6 will described the new improved approach and its working example.approach. And finally author concludes in section 7 with a discussion of future work.

I. REVIEW OF LITERATURE

In this section author has discussed some research papers which had been previously undertaken in the field of frequent pattern mining and erasable itemsets mining.

The Most famous algorithm Apriori for frequent pattern mining was proposed in the year 1999 by Agarwall and Srikanth. Apriori is the Latin word which means 'from the Earlier' means this algorithm uses prior knowledge i.e. knowledge from the previous levels. L. Jing et.al. (2009) have proposed algorithm for improving the efficiency of Apriori algorithm which was done by reducing the number for scanning the database, and reducing the number of candidate itemset. Libing Wu, KuiGong, F. Guo, XiaohuaGe (2010) have also given an improved algorithm which was better than naïve algorithms in time consuming. Zhi-Hong Deng, Guo-Dong Fang, Zhong-Hui (2009) proposed an META algorithm for mining erasable itemsets mining. This algorithm is based on horizontal data layout. Jaishree Singh, Hari Ram, Dr. J.S. Sodhi (2013) had proposed a method for improving efficiency of Apriori Algorithm using transaction reduction.

II. ERASABLE ITEMSET MINING PROBLEM

Erasable itemset mining problem is decomposed into two sub-problems:

1. First sub problem is calculating gain of itemsets which is further decomposed into two sub problems- one is to find the sum of value of all products, second is, to finding the gain of itemsets.

2. Second sub problem is to generate erasable itemsets by using pruning technique for example threshold. Itemsets whose gain is less than the user defined threshold are known as erasable itemsets.

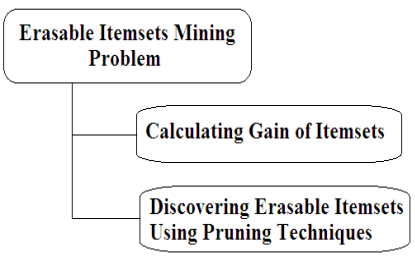

Figure 2 Erasable Itemsets Mining Problem

ISSN 22773061

Let I = {i1, i2, i3, i4…….im} is the set of m different literals known as 'items' also called universal itemset. PDB = {P1, P2, P3, P4 …....Pn} is the product database over I. Means each product contains a set of items {i1, i2, i3, i4,...ij} ⊆ I. and have identifier PID. As shown in example P5 constitute of i6, i7 and have a product identifier PID i.e. 5.

TABLE 1

Product Database

Product	PID	Items	Value
P_1	1	$\{i_2, i_3, i_4, i_6\}$	600
P_2	2	$\{i_4, i_5, i_7\}$	300
P_3	3	$\{i_1, i_4, i_3, i_5\}$	600
P_4	4	$\{i_1, i_2, i_4\}$	8100
P_5	5	$\{i_6, i_7\}$	400
P_6	6	$\{i_3, i_2\}$	600
P_7	7	$\{i_1\}$	400
P_8	8	$\{i_2\}$	500
P_9	9	$\{i_2, i_4\}$	700

There are two main differences in erasable itemset mining and frequent pattern mining, Erasable itemsets mining play important role in production planning in the manufacturing industry. However, FPM focuses on finding the collection of items in the retail trade. Second, Basic computation unit of erasable itemsets is the 'values' of itemsets. Whereas, FPM have 'counts' [2]. Thus erasable itemsets mining and FPM are different mining problems, but many concept of FPM can be applied to erasable itemsets mining as explained in further sections.

III. BASIC CONCEPTION

Pi (i ∈ [1…n]) is a type of product and is represented as: < PID, Items, Val >, PID is the unique identifier of product Pi. Items are items or components that are used to form Pi. Val is gain or profit that a manufacturing industry gets by selling all Pi-type products. Formal definition of Gain of an itemset is [12]:

Definition 1(Gain): Let S (⊆ I) is an itemset (i.e. set of items), the gain of S is defined as

$$\text{Gain}(S) = \sum_{\{Pk \mid S \cap Pk.Items \neq \varnothing\}} Pk.Val \qquad (1)$$

'It means that the gain of itemset S is the sum of profits of all products that are made up of components that constitute at least one item present in itemset S as their components' [2]. For example, for an itemset S (= {i3, i5}). According to equation (1) the gain of S is the sum of P1.Val, P2.Val, P3.Val and P6.Val. Thus gain of S is 2100 (P1.Val (600) + P2.Val (300) + P3.Val (600) + P6.Val (600) =2100).

Definition 2(Threshold): It is the constraint which is used for discovering the erasable itemsets and denoted as 'ξ'. Few algorithms uses percent threshold whereas some use rank threshold for generating the erasable itemsets.

Definition 3 (Erasable Itemsets): Given a user specified threshold (ξ) and a product database PDB, an itemset S is erasable if [12]

$$\text{Gain}(S) \leq \sum_{\{Pk \in PDB\}} (Pk.Val) \times \xi \qquad (2)$$

Thus, the problem of erasable itemsets mining is the problem of searching the itemsets whose gain is less then threshold %.

For example, suppose % threshold is 15%. Then threshold in terms of gain is:

$$\text{Net gain} = \sum_{1 \leq k \leq 9} (Pk.Val)$$

$$= \sum_{\{Pk \in PDB\}} P1.Val + P2.Val + P3.Val + P4.Val + P5.Val + P6.Val + P7.Val + P8.Val + P9.Val \quad (12{,}200)$$

Now according to equation (2), Threshold in terms of Gain is = 12,200 X 15%=1830; Now find itemsets whose gain is less than or equal to 1830. It can be said that itemset {i3} is an erasable, as its gain (1800) is less then threshold that is 1830. Similarly erasable 2-itemset can be obtained, by taking the union of 1-itemset and so on.

IV. META ALGORITHM FOR MINING ERASABLE ITEMSETS

In this section different algorithm related to frequent pattern mining and erasable itemset mining are briefly described META is the abbreviation for Mining Erasable iTemsets using Anti-monotone property. Different algorithms have been proposed for finding erasable itemsets. META is the well known application proposed by Z. Deng, Guo-Dong Fang and Z. Wang (2009) [2]. This algorithm makes use of breadth first search. There are two main steps of this algorithm: first step is, to generate a set of candidate itemsets. Second is to find the gain of each candidate set in database and trim all disqualified candidates.

Terms related to this algorithm are:

Erasable Itemsets: The sets of item which satisfy the constraint (threshold) and it is denoted by Ei for ith itemset.

Apriori Property: Any subset of erasable itemset must be erasable [3].

Join Operation:To find Ek, a set of candidate k-itemsets is generated by joining Ek-1 with itself.

Join Step: candidate item GCk is generated by joining Ek-1 with itself.

Prune Step: META uses two pruning techniques, first is based on the threshold (threshold for erasable itemsets should be less than the user specified threshold) and second is, any (k-1)-itemsets that is not erasable cannot be subset of an erasable k-itemsets [5].

5.1 Pseudo-code for META [8]:

→Scan the product database to find E1, the set of all erasable 1-itemsets, together with their gains;

→For (k=2; Ek-1≠ $\emptyset$; k++) {

-Generate GCk the set of candidate k-itemsets, from Ek-1, the set of erasable (k-1)-itemsets found in the previous step;

-Scan the product database to find the gain of itemsets in GCk;

-Find Ek, a subset of GCk containing k-itemsets with gain less than (ξ (threshold) × Sum of gain of all the products)}

→Return E1 ∪ E2 ∪ E3 ∪ …….∪ Ek;

5.2 Limitations of META:

→ A lot of time spends to deal with large candidate itemsets. And due to large number of records in database results in much more Input/output cost [12].

→ META repeatedly scans the production database to find the gain of candidate itemsets [6].

V. PRINCIPAL FOR IMPROVED ALGORITHM (I-META)

In this, author proposed an optimized method for META algorithm [2] by reducing the size of database. In proposed method, author introduced an attribute Size_Of_Product (SOP), containing number of items in individual product in database. The reduction process of products in database will made according to the value of 'k' and threshold value [4].

6.1 Description of the algorithm:

Algorithm: I-META
Input: product database, PDB = {P1, P2, ….Pn}, a itemsets I, and threshold, ξ.
Output: EI, all erasable itemsets in PDB.
Method: Scan DB to get the overall profit, Sum_val;
Scan DB again to find the set of all erasable 1- itemset, E1;

For (k = 2; Ek-1 ≠ $\emptyset$; k++) {
GCk = Gen_Candidate(Ek-1);
For each product P∈ PDB {
For each candidate itemset C∈ GCk {

If C ∩ P ≠ $\emptyset$ then C.gain= C.value+ P.val;
}
}
Ek = C∈ GCk | C.gain ≤ ξ× Sum_val}
If (k ≥ 2) {
delete_pdatavalue(PDB, Ek, Ek-1);
delete_pdatarow(PDB, Ek); }
}
Return EI = ∪k Ek ;
// Generating candidate itemsets and their PID_lists
Procedure Gen_Candidate(Ek-1: erasable (k-1)-itemsets)

Candidates = $\emptyset$;
For each erasable itemset A1(={x1, x2 , …xk-2, xk-1})∈ Ek-1 {
For each erasable itemset A2(={y1, y2 , …yk-2, yk-1})∈ Ek-1
{//Here, xk-1 < yk-1 means xk-1 is ahead of yk-1 in I .
If ((x1= y1)∧ (x2= y2)∧…∧ (xk-2= yk-2) ∧(xk-1 < yk-1)) then

```
{
X = {x1, x2 , …xk-2, xk-1, yk-1};
If No_Unerasable_Subset(X, Ek-1) then add X to candidates;
}}}
Return Candidates;
Procedure No_Unerasable_Subset(X, Ek-1).
// X: a candidate k-itemset
//Ek-1: the set of all erasable (k -1)-itemsets
For each (k -1)-subset Xs of X {
If Xs ∉ Ek-1 then Return FALSE}
Return TRUE;
Procedure delete_pdatavalue(PDB:database; Ek: erasable K-itemsets;Ek-1: erasable (k-1)- itemsets)
for each itemset I ∉ Ek {
for each product p ∈ PDB {
for each pdatavalue ∈ p{
if(pdatavalue=i) and i ∉ Ek-1
pdatavalue= Ø
}}
Procedure Delete_pdatarow(PDB: database; Ek: erasable(k)-itemsets)
for each product p∈ PDB{
for each pdatavalue ∈ p{
if (pdatavalue!= Ø and pdatavalue != 0 ){
pdatarow.c++ }}
if (pdatarow.c <  k) and pdatarow.Gain> threshold)
delete pdatarow }
```

6.2 Example of algorithm:

Product database PDB is shown in Table 1. Suppose the constraint threshold is 20% i.e. 2440. The algorithm is as follows:

Step 1: Convert database into desired database by adding column SOP

Step 2: In the first iteration of the algorithm, each item is a member of the set of candidate 1-itemsets, GC1. The algorithm simply scans the whole product database in order to calculate the gain of each item.

Step3: This algorithm will then count the number of items in each product known as Size_Of_Product (SOP).

Step 4: As threshold 20%, the erasable 1-itemset, E1 can be determined. Then from this find GC1.

Step 5: Gain of i1, i2 and i4 is more than threshold (2440), these does not appear in E1. Delete these data items from D. and also delete those records of transaction having SOP=1in D and items are not in the previous erasable itemset. Thus delete P7, P8 and P9.

Step 6: Delete the record with Value is greater than threshold i.e. product P4 from D and obtain D1.

Step 7: To find the set erasable 2-itemset, E2, algorithm uses the join E1∞E1 to generate candidate set of 2-itemset, GC2.

Step 8: D1 is scanned to find the gain of GC2.

Step 9: Determine the erasable 2-itemset, E2, having gain greater than the threshold.

Step 10: Now from the erasable itemset E2 find GC3 by applying the join E2∞E2. And remove the itemsets from this result which are not in the E2, according to the downward closure property.

Step 11: database D2 is scanned to calculate the gain of itemsets in GC3. And determine E3 from GC3 by comparing its gain with threshold.

Step 12: E3 has only one 3-itemset so GC4=Ø, the algorithm will stop here. This algorithm generate GCk until GC(k+1) become empty.

D

PID	Itemsets	Val
1	{i_2, i_3, i_4, i_6}	600
2	{i_4, i_5, i_7}	300
3	{i_1, i_4, i_3, i_5}	600
4	{i_1, i_2, i_4}	8100
5	{i_6, i_7}	400
6	{i_3, i_2}	600
7	{i_1}	400
8	{i_2}	500
9	{i_2, i_4}	700

Scan D to count SOP

Scan D to find gain

Itemset	Gain
i_1	9100
i_2	10500
i_3	1800
i_4	10300
i_5	900
i_6	1000
i_7	700

Compare gain With threshold

PID	Itemsets	Val	SOP
1	{i_2, i_3, i_4, i_6}	600	4
2	{i_4, i_5, i_7}	300	3
3	{i_1, i_4, i_3, i_5}	600	4
4	{i_1, i_2, i_4}	8100	3
5	{i_6, i_7}	400	2
6	{i_3, i_2}	600	2
7	{i_1}	400	1
8	{i_2}	500	1
9	{i_2, i_4}	700	2

E1

GC2

Itemsets	Gain
i_3	1800
i_5	900
i_6	1000
i_7	700

Generate From E1

Itemsets
{i_3,i_5}
{i_3, i_6}
{i_3, i_7}
{i_5, i_6}
{i_5, i_7}
{i_6, i_7}

GC2

ISSN 22773061

D1

PID	Itemsets	SOP
1	$\{i_3, i_6\}$	2
2	$\{i_5, i_7\}$	2
3	$\{i_3, i_5\}$	2
4	-	-
5	$\{i_6, i_7\}$	2
6	$\{i_3\}$	1

Scan D to Find Gain of GC2

Itemsets	Gain
$\{i_3,i_5\}$	2100
$\{i_3, i_6\}$	2200
$\{i_3, i_7\}$	2500
$\{i_5, i_6\}$	1900
$\{i_5, i_7\}$	1300
$\{i_6, i_7\}$	1300

Compare Gain with Threshold to find E2

E2

Itemsets	Gain
$\{i_3, i_5\}$	2100
$\{i_3, i_6\}$	2200
$\{i_5, i_6\}$	1900
$\{i_5, i_7\}$	1300
$\{i_6, i_7\}$	1300

Generate GC3 from E2

Itemsets
$\{i_3, i_5, i_7\}$
$\{i_5, i_6, i_7\}$

D2

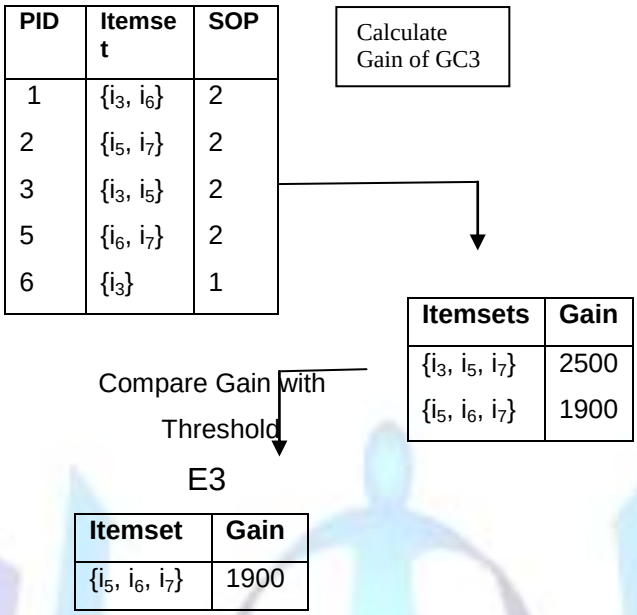

Figure 3 Example of Algorithm

VI. CONCLUSION

In Last few years, a lot of people have given several algorithms and compare different algorithm to solve the erasable itemset problem as efficiently as possible. In this Paper, META algorithm is improved by using the property 'reduction in product database'. The classical META algorithm has performance bottleneck in large databases. Thus it is very important to improve the performance algorithm with various methods. In this new approach the author give the key idea of reducing time. Future Scope: This idea of database reduction will definitely open new scopes for young researchers to work in the field of data mining. Although, this improved META is efficient but it has overhead to manage the new database. So there should be some solution which has less number of scans of whole product database. For future, researcher can divide the large database among processors to reduce the time for database scan.

REFERENCES

[1] Sanjeev Rao, Prianka Gupta, "Implimenting Improved Algorithm Over APRIORI Data Mining Association Rule Algorithm", In: proceeding of IJCST, VOL.3, Issue 1, Jan-March 2012.

[2] Deng, Z., Fang, G., Wang, Z., Xu, X., "Mining Erasable Itemsets". In: 8th IEEE International Conference on Machine Learning and Cybernetics, pp. 67–73. IEEE Press, New York (2009).

[3] Sunita B.Aher, Lobo L.M.R.J.,"A Comparative Study of Association Rule Algorithms for Course Recommender System in E-learning", In: Proceeding of IJCA, Vol 39-No.1, February 2012.

[4] Jaishree Singh, Hari Ram, Dr. J.S. Sodhi, "Improving Efficiency of Apriori Algorithm Using Transaction Reduction", In: proceeding of

IJSRP, Volume 3, Issue 1, January 2013.

[5] Badri Patel, Vijay K Chaudhary, Rajesneesh K Karan, YK Rana, "Optimization of Association Rule Mining Apriori Algorithm Using ACO", In: Proceeding of IJSCE, Volume-1, Issue-1, March 2011.

[6] Shruti Aggarwal, Ranveer Kaur, "Comparative Study of Various Improved Versions of Apriori Algorithm", In: proceeding of IJETT, Volume 4, Issue 4,April-2013.

[7] K. Geetha, Sk. Mohiddin, "An Efficient Data Mining Technique for Generating Frequent Item Sets", In: Proceeding of IJARCSSE, volume 3, Issue 4, April 2013.

[8] Partibha Parikh Dinesh Waghela, "Comparative Study of Association Rule Mining Algorithms", In: Proceeding of UNIASCIT, Volume 2, Isuue 1, 2012.

[9] K. Vanitha, R. Santhi, "Evaluating the Performance of Association Rule Mining Algorithm", In: proceeding of JGRCS, Volume 2, Issue 6, June 2011.

[10] Rakesh Agrawal, T. Imieliński, A. Swami, "Mining association rules between sets of items in large databases". In: Proceedings of the 1993 ACM SIGMOD international conference on Management of data - SIGMOD '93. pp. 207-216.

[11] Zhihong Deng and Xiaoran Xu, China, “Mining Top-rank-K Erasable Itemsets”, in ICIC Express Letters, ICIC International @ 2011 ISSN 1881-803X, Volume 5, Number 1, January 2011.

[12] Zhihong Deng and Xiaoran Xu, “An efficient Approach for Mining Erasable Itemsets”, in ADMA 2010, Part I, LNCS 6440, pp. 214–225, 2010.

AUTHORS

Miss. Shweta received the B. Tech degree in computer science and engineering from Kurukshetra University in 2011. She is now pursuing M. Tech in computer science from department of computer science and application at the Kurukshetra University, Haryana. This author has published one review paper at national level and one research paper in international journal. Her research interest includes data mining.

Dr Kanwal Garg presently working as Assistant Professor in Department Of Computer Science And Application, Kurukshetra University Kurukshetra. Apart from district topper in senior secondary examination and Kurukshetra University topper in under graduation examination, he had completed his post graduation & doctorate from GJU S&T, Hisar under the faculty of Engineering and Technology. Owe the credit of more than 45 research papers published in international & national journals, conference & seminar. He attended 12 workshops/faculty Development Programme/winter/summer school and 01 Orientation Programme to enhance his curriculum. His area of expertise is Data Bases, Data Mining, & warehousing. Approximately 11 year of experience in teaching industry and administration. During this tenure he is actively involved in organizing co-curricular & social activities.