International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-6, April 2020

Mining Frequent Itemsets over Uncertain Database using Matrix

Deepak Kumar Sharma, Samar Wazir, Md. Tabrez Nafis, Amit Kumar

Abstract: ***In the area of data mining for finding frequent itemset from huge database, there exist a lot of algorithms, out of all Apriori algorithm is the base of all algorithms. In Uapriori algorithm each items existential probability is examined with a given support count, if it is greater or equal then these items are known as frequent items, otherwise these are known as infrequent itemsets. In this paper matrix technology has been introduced over Uapriori algorithm which reduces execution time and computational complexity for finding frequent itemset from uncertain transactional database. In the modern era, volume of data is increasing exponentially and highly optimized algorithm is needed for processing such a large amount of data in less time. The proposed algorithm can be used in the field of data mining for retrieving frequent itemset from a large volume of database by taking very less computation complexity.***

Keywords : Certain Transactional Database, Uncertain Transaction Dataset, existential probability, Matrix, Data mining, machine learning.

I. INTRODUCTION

The data in the modern world increasing exponentially due to which it attracts many researchers to contribute with effective algorithms which scan huge amount of data to discover useful information which is known as Knowledge Discovery (also known as Data mining). Frequent Itemset Mining (FIM) is a method in which frequent itemsets are generated from the database using a minimum support; the itemset which is greater than minimum support is known as frequent itemsets. Association rule is a machine learning process which is used to revile an interesting relationship between items in the large datasets. Let us consider S={s1,s2,.....,sn} be a set of 'n' items and U={u1,u2,...,un} be a set of transactions. P->Q where P and Q are set of items in S. This means that if P satisfy then there is a high probability that Q also satisfy.

a) Let P and Q are two distinct item in the transaction U, then the Support of Association rule can be given as the ratio of records that contain X U Y to the total number of records in the database. The formula to get the support is as follows:

Support(P|Q) =Support of PQ / Total transactions

b) Confidence can be defined as the ratio of the records that holds P U Q to the total number of transaction that holds P, confidence is directly proportional to strong association rule i.e. P=>Q.

Confidence(P|Q) =Support count of PQ / Support count of P

There are two types of database:-

- Certain Transactional Database (CTDB): In such type of database each transaction has some items. For example, a customer purchases a Notebook and pencil. Let, T1 = { notebook, pencil } i.e. if someone who purchases notebook then there is a very high probability to purchase pencil.
- Uncertain Transaction Dataset (UTDB): in such type of dataset transaction consists of items with existential probability with it. For example ,T2 = {notebook (0.60), pencil (0.80) }i.e there is 60% chance to buy a notebook. If someone purchases notebook then there will be 80% chance that he will buy pencil. This uncertain buying pattern of customer for each item in the transaction is also termed as Attribute uncertainty[1,2].

In the modern year of data explosion (means day by day the amount of data is increasing exponents), it is very important to develop efficient algorithms to handle efficiently such huge amounts of data with respect to memory, CPU and I/O cost[1].

In this paper, we are going to use a matrix structure for generating 1 and 2 frequent itemset directly. After 2 frequent itemset the algorithm will user UTDB for generating Lk-1 frequent itemsets. In this proposed algorithm matrix structure is used as a meta data for generating 1 and 2 frequent itemset very quickly. The main problem with any step by step approach is to generate and handle 2- itemsets. If there are 500 items in a dataset then there will be 2500-1 itemsets in total. In our proposed algorithm we define a matrix of 500x500 array of 2-D which will hold all the combination of 1 and 2 itemset information, and this matrix can be used directly to get 1 and 2 frequent itemset. This approach reduces the time and CPU computational in the frequent itemset mining process.

II. RELATED WORKS

Frequent itemset mining (FIM)[3,22] is a technique for generating frequent itemsets (hence occurred frequent) in a large database. FIM basically tells how often a set of items is being purchased together. Apriori algorithm is the oldest algorithm which is used for extracting Frequent itemsets. Apriori algorithm was first introduced by R. Agrawal in 1993 [4] and it uses the iterative approach for finding FIM.

Revised Manuscript Received on April 30, 2020.

***** Correspondence Author

Deepak Kumar Sharma*, (Pursuing M. Tech) Computer Science & Engineering, Jamia Hamdard University, New Delhi, India. E-mail: deepakks222@gmail.com

Dr. Samar Wazir, Computer Science & Engineering, Jamia Hamdard University, New Delhi, India. E-mail: samar.wazir786@gmail.com

Md. Tabrez Nafis, Computer Science & Engineering, Jamia Hamdard University, New Delhi, India. E-mail: tabrez.nafis@gmail.com

Amit Kumar, (Pursuing M. Tech) Computer Science & Engineering, Jamia Hamdard University, New Delhi, India. E-mail: muskanyadav.sanpla@gmail.com

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC-BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

This algorithm uses certain transactional database (Example of Transactional database, Let, T1={noteBook, pencil, rubber}, here noteBook, pencil, and rubber are the name of an item in transaction 1) for extracting FIM. This algorithm uses bottom-up methodology for extracting useful information. At the first step, entire frequent itemset are generated who satisfy the thrushold by scanning the full database. The same process continues until there is no more frequent K- itemsets is possible. The problem with this algorithm is that , as the dataset is very large, this approach is not efficient and requires large number of scan of dataset [2]. The limitation of Apriori has been removed in FP growth algorithms which were introduced by J.Han in 2000 [5] and it is one of the most popular algorithm used for extracting FIM from the transactional database which uses divide and conquer technique. It uses an FP tree structure, which uses vertical and horizontal database layout for finding FIM. At first, scanning of the database is done then we get frequent items 'A' with there support. After that 'A' is sorted in descending order w.r.t support. Lastly, create a 'root' of a tree.Thereafter, Equivalence Class Transformation(EclaT) algorithm has been proposed by Zaki[6] in 2000 for mining frequent itemsets using vertical data format . Later, modified Apriori algorithm has been proposed by Thanda Tin Vn [7] and it uses a matrix structure to generate frequent itemset. This algorithm can only generate FIM from a certain transactional database. Apriori, FP growth and modified Apriori algorithms uses certain transactional database for generating FIM.

Uapriori is the first algorithm introduced by chun Kit Chui [8] in 2007 which is used to generate FIM by using uncertain transactional database(Example of uncertain database, let T1={notebook (0.4), pencil (0.8) and rubber (0.4) }, here notebook, pencil and rubber is the name of the item whereas (0.4),(0.8) and (0.4) is the existential probability associated with each item respectively in transaction-1). Uapriori also works on the same principle of the Apriori algorithm(i.e step by step), so, it also suffers from the same problem as generation and handle of huge Candidate itemsets. Many algorithms are proposed for FIM over UTDB [9,10,12-18] after Uapriori. To overcome this problem, the UFP growth algorithm was introduced, which uses the same feature of FP growth , and generate FIM without generating candidate itemsets.

III. PROBLEM STATEMENT

In any step by step approach for frequent itemset, minings has to cater a large candidate set. For example, if there are K single items then the total number of itemset generated is m=2K-1 [19]. The main problem with this type of approach is to generate and process 2 itemsets. In our proposed algorithm, we overcome the limitation of generating 2 itemsets by using a matrix approach.

IV. PROPOSED SOLUTION

Algorithm steps are as follows:-

Inputs of the algorithm:

- 'D' UTDB

Table- I: Uncertain transactional database (D)

TID	Items				
1	1(0.4)	2(0.5)		4(0.4)	5(0.6)
2		2(0.4)	3(0.3)		5(0.3)
3	1(0.4)	2(0.5)		4(0.2)	5(0.6)
4	1(0.8)	2(0.6)	3(0.5)		5(0.8)

- Support count: minimum threshold (Ex: 0.5)

Step 1) Count the distinct items (n) in D, as we can see from Table 1 n=5.

Step 2) Make a matrix 'a' with (n x n) dimensions, i.e. a=[5][5] and and initialized all the elements of matrix with zero.

Table- II: Initialized all element of a[5][5] with zero

	1	2	3	4	5
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	0	0	0	0	0
5	0	0	0	0	0

Step 3) Filling up the matrix 's' from the transactions in Table I.

For transaction T1 , matrix is filled up for one and two itemset combinations-

T1 = {1(0.4), 2(0.5),4(0.4),5(0.6)}

Table- III: : Combination of 1 and 2 itemsets with there existential probability

Items	Existential Probability
1	0.4
2	0.5
4	0.4
5	0.6
1,2	0.4*0.5=0.20
1,4	0.4*0.4=0.16
1,5	0.4*0.6=0.24
2,4	0.5*0.4=0.20
2,5	0.5*0.6=0.30
4,5	0.4*0.6=0.24

Table- IV: : Combination Matrix 'A' filled with transaction T1

	1	2	3	4	5
1	0.4	0.20	0	0.16	0.24
2	0	0.5	0	0.20	0.30
3	0	0	0	0	0
4	0	0	0	0.4	0.24
5	0	0	0	0	0.6

Similarly by mapping T2,T3 and T4 in the matrix, we get the final matrix as following:-

Table- V: Matrix 'A' filled with all the transaction

	1	2	3	4	5
1	1.6	0.88	0.40	0.24	1.18
2	0	2.0	0.42	0.30	1.20
3	0	0	0.8	0	0.09
4	0	0	0	0.6	0.36
5	0	0	0	0	2.3

Step 4) Now, we will generate the frequent item with the help of Table V matrix as follows:

begin Procedure UAprioriFIM(UDB, PL)

L2= Matrix_file(matrix A,PL)

/*Frequent item set of level 1and 2 is generated have supportcount> PL */

For (a=3; La-1 ≠Ø ; a++)

CK=UAprioriFIM(La-1-1)

// Candidates set

M_K= { c Ɛ Ca | c.expectedSupport>=PL }

End

Begin Matrix_file(matrix A, PL)

For (c=1;c<=A.size;c++)

For (d=c;dj<=A.size;d++)

If (a[c][d]>PL)

L=L U a[c][d]

End if

End for

End for

Return L;

End

For example the output of the algorithm (using Thrushold value=0.5) is as follows:

Table- VI: All frequent itemset generated for database 'D'

Frequent Itemsets of 1 items generated using matrix	1(1.6),2(2.0),3(0.8),4(0.6), 5(2.3)
Frequent Itemsets of 2 items generated using matrix	{(1,5)(1.18)}, {(2,5)(1.20)}
Frequent Itemsets of 3 items generated using Uapriori	Nil

V. EXPERIMENT AND ANALYSIS OF RESULT

To determine the efficiency and performance of the modified Uapriori Algorithm using Matrix, All the experiments are executed on Amd8 Processor having 4GB of Random Access Memory.

We use the 20K,40K,80K,320K,Gazelle and Accident dataset [20] as an uncertain database and both the algorithm are run over these datasets (Uapriori [21] is mendioned as Original and Modified Uapriori is mendioned as Modified algorithm) with different thrushold values. The observations obtained on these data set are outlined in Table 7. As we can find out from the result that modified Uapriori takes less time with compare to the original Uapriori algorithm as it uses matrix freature, and it generate the 1 and 2 itemset directly from the created matrix. For reset of the itemset is generated by the dataset.

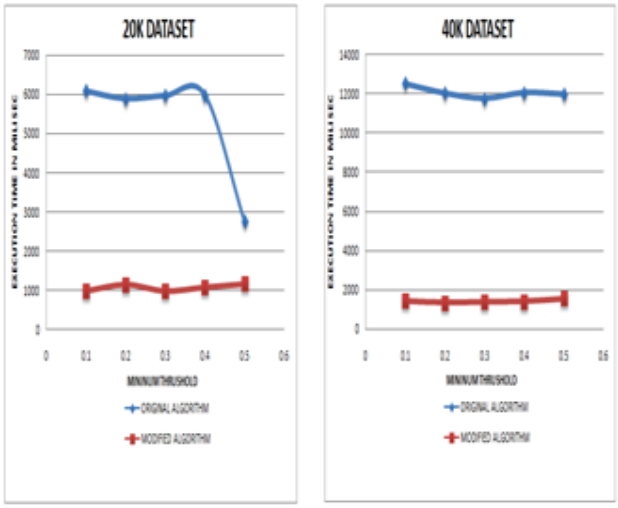

Fig. 1.Execution of Uapriori and Proposed algorithm on 20K dataset

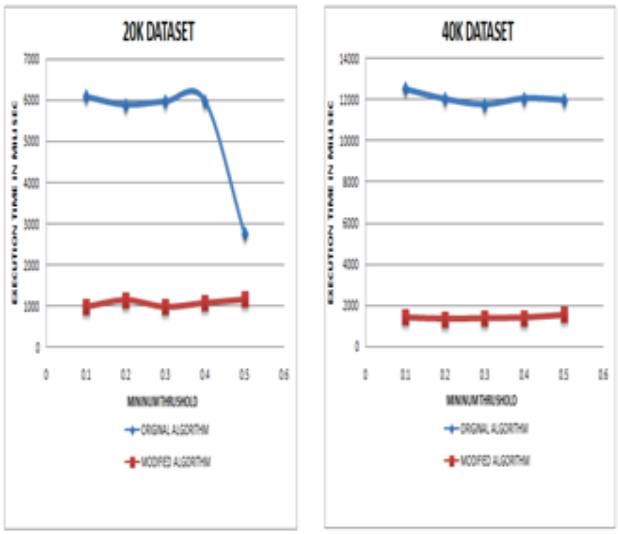

Fig. 2.Execution of Uapriori and Proposed algorithm on 40K dataset

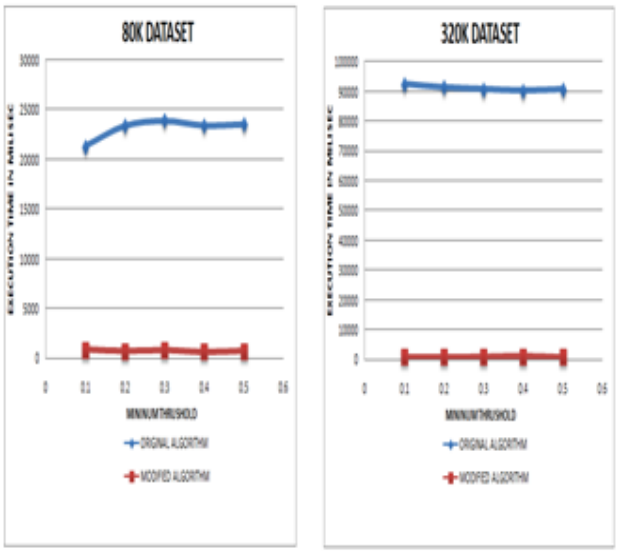

Fig. 3.Execution of Uapriori and Proposed algorithm on 80K dataset

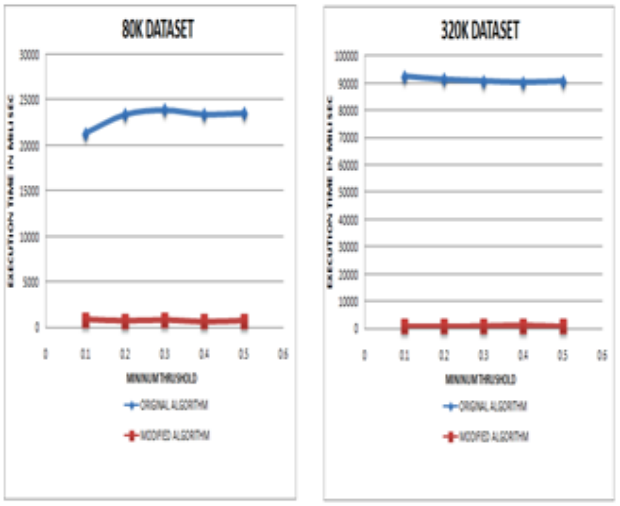

Fig. 4.Execution of Uapriori and Proposed algorithm on 320K dataset

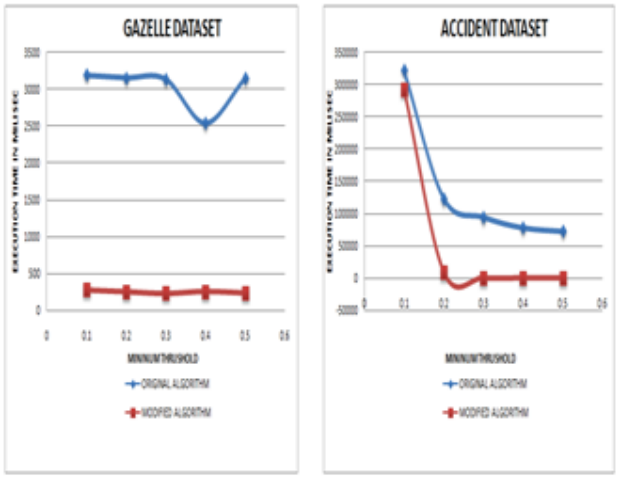

Fig. 5.Execution of Uapriori and Proposed algorithm on Gazelle dataset

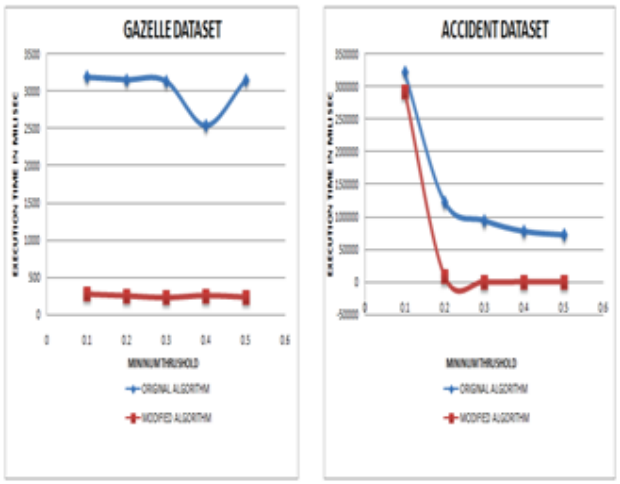

Fig. 6.Execution of Uapriori and Proposed algorithm on Accident dataset

VI. CONCLUSION

The proposed algorithm which is using matrix technology is highly effective and highly optimized to generate the desired result by processing a huge volume of database. The proposed algorithm has been executed on different size of uncertain transactional database to generate frequent items which takes very less computational time as compared to the traditional algorithm. So, it will become a stepping stone in the area of data mining, where the retrival of usefull information from a large database is always a paramount challenge and a daunting task.

In the process of this paper, we observed that increasing the matrix dimensions will give even better result. In the future, we may achive better result by decreasing computational time for getting usefull information by increasing the dimentional of matrix.

But to increase the dimensional of matrix, the proposed algorithm needs to be modified.

REFERENCES

1. Jamsheela O, Raju G (2015) Frequent itemset mining algorithms: a literature survey. In: Paper presented at the 2015 IEEE international advance computing conference (IACC), Banglore.
2. Mamta Dhanda, Sonali Guglani, “Mining Efficient Association rules Through Apriori ”, In: Proceeding of IJCST, ISSN0876-8491, Vol. 2, Issue 3, September.
3. S. Ruggieri, Frequent regular itemset mining, Knowledge Discovery and Data Mining (KDD) (2010) 263–272.
4. R. Agrawal, R. Srikant, Fast algoritms for mining assocaitation rules, in Proceeding of the 20th VLDB Conference, Santiago, Chile (1994) pp. 487,499.
5. Jiawei Han, Mining Frequent Patterns without Candidate Generation: A Frequent Pattern Tree Approach, in Data Mining and Knowledge Discover, 8,53-87,2004.
6. J. Zaki, Scalable Algorithms for Association Mining, in IEEE transations on Knowledge and Data Engineering, vol. 12, No. 3, 2000.
7. Thanda Tin Yu, Proposed Method for Modified Apriori Algorithm, in Int’l Conf. Information and Knowledge Engineering | IKE’17.
8. C.K. Chui, B. Kao, E. Hung, Mining frequent itemsets from uncertain data, in 11th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining, PAKDD 2007, Nanjing, China.
9. T. Bernecker, R. Cheng, H.P. Kriegel, M. Renz, F. Verhein, A. Zufle, D.W. Cheung, S.D. Lee, Wang Liang, Model-based probabliistic frequent itemset mining. Knowl. Inf. Syst. 37,181-217(2013).
10. T. Bernecker, H.-P. Kriegel, M. Renz, F. Verhein, A. Züfle, Probabilistic frequent itemset mining in uncertain databases in Proceedings of the 15th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD’09), Paris, France.
11. R. Cheng, D. Kalashnikov, S. Prabhakar, Evaluating probabilistic queries over imprecise data, in SIGMOD (2003).
12. Q. Zhang, F. Li, K. Yi, Finding frequent items in probabilistic data, in SIGMOD (2008).
13. L. Sun, R. Cheng, D.W. Cheung, J. Cheng, Mining uncertain data with probabilistic guarantees, in SIGKDD (2010).
14. C.K. Chui, B. Kao, A decremental approach for mining frequent itemsets from uncertain data, in PAKDD (2008), pp. 64–75.
15. Y. Tong, L. Chen, Y. Cheng, P.S. Yu, Mining frequent itemsets over uncertain databases, in VLDB’12.
16. Y. Tong, L. Chen, P.S. Yu, UFIMT: an uncertain frequent itemset mining toolbox, in KDD’12, Beijing, China. 12–16 August 2012.
17. Garg, R., Nafis, M. T. Nafis, & Garg, B. (2018). Ordered weighted averaging operator used to enhance the accuracy of fuzzy predictor based on genetic algorithm. International Journal of Intelligent System Technologies and Applications 17(1-2), 229-253.
18. Nafis, M.T., & Biswas, R, (2018). A Secure Clustering Technique for Unstructured and Uncertain Big Data. In Progress in Advanced Computing and Intelligent Engineering(pp. 459-466)(Springer).
19. Alberto Cano, Speeding Up Association Rule Mining with Inverted Index Compression, in IEEE Transactions on Cybernetics.
20. Itemset Mining Implementions Repository, http://fimi.ua.ac.be/
21. SPMF An Open Source Data Mining Library, http://www.philippe-fournier-viger.com/spmf/index.php?link=datasets.php
22. Charu C Aggarwal, Frequent Pattern Mining with Uncertain Data, in International Conference on Knowledge Discovery and Data Minign,Paris, France, June 28- July 1,2009.

AUTHORS PROFILE

Deepak Kumar Sharma obtained his Bachelor in Computer Application (from North Bengal University) and Master in Computer Application (from Sikkim Manipal University) degree. He is presently he is pursuing M.Tech in Computer Science at the Department of Computer Science and Engineering, Jamia Hamdard University, New Delhi, India. Currently he is working as Senior Database programmer at Government sector. While performing his normal duty he got a certificate of Excellence in participated in digitalization of india on behalf of Prime Minister of India. His area of interest is Data Mining, Artificial Intelligent, Deep Learning and Machine Learning.

Dr Samar Wazir obtained his B.Tech degree in Information Technology, M.Tech in Computer Science and PhD in Computer Engineering. Currently he is working as Assistant Professor at the Department of Computer Science and Engineering at the Jamia Hamdard University and also serving as University NCC officer. His area of interest is Association Rule Mining. He has published many papers on a wide range of topics in Frequent Itemset Mining, Probablity, Fuzzy Theory and OWA.

Dr Md Tabrez Nafis is an Assistant Professor at the Department of Computer Science and Engineering, Jamia Hamdard University, New Delhi, India. Dr. Nafis has published several research papers in reputed International Journals and Conferences. Dr Nafis is a member of several International/National professional bodies viz. IEEE, CSI, IETE, ISTE. His research interest areas are Big Data, Data Science, Machine Learning.

Amit Kumar obtained his Bachelor in Computer Application (IGNOU) and Master in Computer Application (IGNOU) degree. He is presently pursuing M.Tech in Computer Science at the Department of Computer Science and Engineering, Jamia Hamdard University, New Delhi, India. Currently he is working as Senior Web and database programmer at Government sector. While performing his normal duty he participated in digitalization of india project. His area of interest is Data Mining, Artificial Intelligent and Machine Learning.