Association Rule Generation in Data Streams Using Apriori Algorithms

S.Vijayarani , R.Prasannalakshmi

Abstract: **Data mining technology is employed for locating useful and unknown knowledge from the massive databases. Normally, data mining techniques are applied to static databases for knowledge extraction whereas the current data mining techniques are not suitable and it also has some limitations for handling dynamic databases. A data stream handles dynamic data sets and it has become one of the important research domains in data mining. The basic definition of data stream is arrival of continuous, ordered and large quantity of data. In order to perform data analysis, finding the relationships between the data and extracting knowledge from the data stream is very difficult because the existing data mining techniques are not adequate. Hence, this situation has raised concerns about the development of new algorithms and techniques for handling data streams. Important data mining tasks performed in data streams are clustering, classification, generation of association rules and frequent item mining. Association rule mining is one of the popular research problems in data stream which helps to find out interesting relations between the data items in the transactional databases. This research work mainly focused on how the traditional algorithms are used for generating association rules in data streams. The algorithms used in this work are APRIORI, APRIORI PT and APRIORI MR. The performance measures used for finding the best algorithm is execution time and number of rules generated. From the experimental results it is observed that APRIORI MR algorithm's efficiency is better than APRIORI and APRIORI PT Algorithms. This work is implemented in Tanagra data mining tool.**

Intex Terms **- Data mining, Data Stream, Association Rules, Apriori, Apriori PT, Apriori MR, Tanagra.**

I. INTRODUCTION

Data stream is a continuous arrival of data which is unlimited in nature. The main characteristics of data stream is it handles prime volumes of continuous data and most probably infinite. Applications areas of data streams are market-basket information analysis, cross-marketing, catalogue style, loss-leader analysis, business organizations (process credit card transactions), financial markets (stock replacements), engineering and industrial processes (power supply and manufacturing), security (traffic engineering monitoring) and web (web logs and web page click streams).Important data mining tasks performed in data streams are clustering, classification, association rule generation, query optimization and frequent item set mining [1].

Manuscript received Dec 26, 2014.

S.Vijayarani, School of Computer Science and Engineering, Department of Computer Science, Bharathiar University, Coimbatore, India

R.Prasannalakshmi, School of Computer Science and Engineering, Department of Computer Science, Bharathiar University, Coimbatore, India

Association rules are defined by finding the frequent patterns, links, correlation and the relevant structures among the data objects in the databases and information repositories. There are two important steps in association rule mining, first one is to find the frequent data items and the second step is to generate association rules using these frequent data items. Association rule mining problem is stated as, consider a given set of items $I=\{I_1,I_2,\ldots I_m\}$ and a database of transactions $D=\{t_1,t_2,\ldots t_n\}$ where $t_i=\{I_{i1},I_{i2},\ldots .I_{ik}\}$ and $I_{ij} \in I$, an association rule is an implication of the form $X \Rightarrow Y$ where $X,Y \subset I$ are sets of items called itemsets and $X \cap Y=\theta$[2]

Two important measures support and confidence are used for association rule generation. The support of an item (or set of items) is the % of transactions in which that item (or items) occurs. The support (s) for an association rule $X \Rightarrow Y$ is the percentage of transactions in the database that contain $X \cup Y$. The confidence or strength (α) for an association rule $X \Rightarrow Y$ is the ratio of the number of transactions that contain $X \cup Y$ to the number of transactions that contain X. Normally, confidence measures the strength of the rule, whereas support measures how often it should occur in the database [6]. Some of the important association rule mining algorithms are apriori, fp-tree, fp-growth, dynamic item set counting, ECLAT, DCLAT and RARM [3]

This research work primarily focuses on generating association rules from data streams. The continuous arrival of data is partitioned and it is stored in the databases. For each and every partition, association rule generation algorithms are applied to generate the association rules. In this work, the traditional association rule algorithms namely Apriori, Apriori PT and Apriori MR are used for generating association rules in each partition. From this, we come to know that the advantages, drawbacks and limitations of these traditional association rule mining algorithms for generating association rules in data streams.

The remaining portion of this paper is organized as follows. Section 2 gives the review of literature. Proposed methodology and the traditional association rule algorithms are described in Section 3. Section 4 discusses experimental results and conclusion is given in Section 5.

II. LITERATURE REVIEW

S.Vijayarani,R.Prasannalakshmi discussed about frequent item mining from the data streams. Eclat association rule mining algorithm is used for frequent item mining. Dataset is partitioned into several windows and each partition, different thresholds values are applied and the Eclat algorithm identified the frequent items in each window. The performance factors used in this work are number of frequent items generated and the execution time [12].

Charu C. Aggarwal. provided the detailed information about data streams. He also discussed how to apply different data mining technologies to data streams for useful and hidden knowledge extraction. He explained data stream clustering, data stream classification and data stream frequent pattern mining in a detailed manner and also the algorithms which are required to perform these tasks are also discussed [3].

Charanjeet Kaur defined how to generate association rules using association rule mining algorithms particularly apriori algorithm. This paper gives the information about the basic concepts of association rule mining, measures used for generating frequent item set. Author has analyzed various types of apriori algorithms like an improved apriori algorithm, distributed apriori association rule, apriori algorithm using ant colony optimization, an improved apriori algorithm based on pruning optimization and transaction reduction [5].

Nan Jiang and Le Gruenwald presented various research issues in data streams. Authors also discussed the general issues in data stream association rule mining like data processing model, memory management, i.e. how an information is collected and stored in memory, how to develop efficient and compact data structures for handling data streams and the need for development of one pass algorithm for generating association rules. They also discussed various application dependent issues [13].

III. PROPOSED METHODOLOGY

The system architecture of the proposed work is represented in Figure 1.

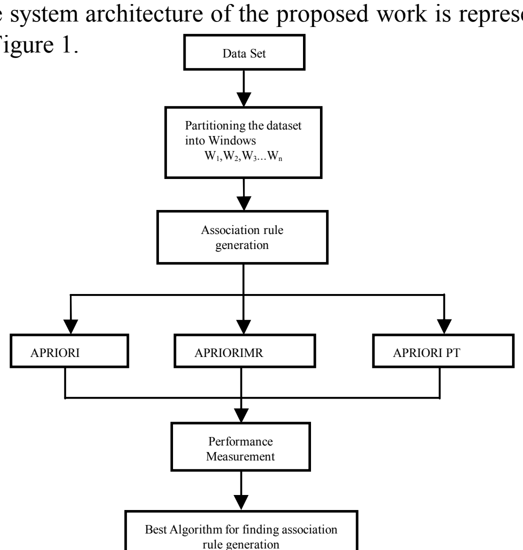

Fig. 1 System Architecture

A. Dataset:

The connect data set is used in this work. It is extracted from http://fimi.ua.ac.be/data/connect.dat. It consists of 67,558 instances and 48 attributes. From this 1K, 2K and 5K instances are used in this work.In data streams, we imagine that the continuous arrival of data is partitioned into several windows with fixed size, i.e. $W_1, W_2, W_3 \ldots W_n$.In this work, we have created five windowsW_1, W_2, W_3, W_4, W_5 with the fixed data set size of 1K, 2K and 5K [9].

B. Association Rule Generation

In order to generate association rules, three types of apriori algorithms are used [11]

- ✓ APriori Algorithm
- ✓ APriori MR – Apriori Map/Reduce Algorithm
- ✓ APriori PT – Apriori Prefix Tree Algorithm

C. APRIORI Algorithm

It is one of the popular and basic association rule mining algorithm. For example, to given a threshold C, the APRIORI formula identifies the item sets that as subsets of a minimum of transaction within the data bases. APRIORI uses a bottom up methodology, where frequent subsets measure extended one item at a time (a step referred to as candidate generation), and groups of candidates are unit tested against the data. APRIORI uses breadth-first search and a hash tree structure to count the candidate item sets efficiently. It generates the candidate item sets of length k from item sets of length k-1. Then, it reduces the candidates that have associated occasional sub pattern. Keep with the downward ending lemma, the candidate set contains all frequent k-length item sets. The pseudo code of this algorithm is given in Table 1 [4].

Table 1. Pseudo Code for APRIORI

1. Join Step : C_k is candidate generated by joining of L_{k-1} with itself
2. Prune Step: Any of k-1 itemset is a not frequent item set cannot be a subset of a frequent k-itemset
3. Pseudo-Code:
4. C_k = size of candidate item set k
5. L_k = size of frequent item set k
6. L_1 = frequent items
7. Loop begins, For (k=1; L_k!= 0; k++) do
8. C_{k-1} = candidate generate to L_k
9. for each transaction t in a database d do
10. All candidate increments and count the C_{k-1}
11. That are included in transaction t
12. L_{k-1} = Min_Support of Candidates in C_{k-1}
13. End
14. Return $U_k L_k$;

Algorithm works as follows,

- ✓ Let frequent item sets (item sets that have minimum support) = F_k containing concepts C_k where size of item sets=k
  - o The first scans of the database and searches for frequent item sets and count for each item.
  - o Then, it measure up to item sets with minimum support required.
  - o It then shows again the following steps to extract all item sets.

- Generate C_{k+1} candidate of frequent item sets of size K+1.

✓ Sets of size k.

- Scan the database as above.
- Add the item sets that assure minimum support requirement.

D. APRIORI MR Algorithm

Apriori-Map/Reduce algorithm runs on parallel Map/Reduce framework. Candidate generation of Apriori Map/Reduce algorithm is prune(C_{k+1}) function is to remove the non-frequent item set C_{k+1} by eliminating non-frequent item sets C_k as non-frequent item sets cannot be a subset of frequent item sets. Table 2 represents the apriori MR algorithm.

Table2. Pseudo Code for APRIORI Map/Reduce Algorithm

1. Map transaction t in a data supply to all Map nodes
2. Each Map node can handle m
3. Now, can use Candidate Map Cm_1 = size of 1 is frequent item set at the node m
4. Reduce and compute candidate generation of C_1 and L_1 with all Cm_1
5. C_1 = size one of frequent item sets;
6. Calculate the Min_Support = Num/ total items;
7. Size 1 of frequent item sets Min_Support is L_1
8. Loop begins, For (k=1; L_k!= 0; k++) do
9. Each mapped node m is represent by L_k . Such as, L_{mk}
10. Sort and remove the duplicate item sets
11. Can use, $C_{m(k+1)} = L_k$ join_sort L_{mk} ;
12. Reduce methods to use the APRIORI Property to computes the C_{k+1} do
13. Each map node m is increment the count of $L_{m(k+1)}$ candidates. That are supplied by transaction t
14. End.
15. Now, Can use reduce method to find the L_{k+1} with $L_{m(k+1)}$ and Min_Support.
16. Min_Support of frequent item set generated by size of k+1 is L_{k+1}.
17. End
18. Return $U_k L_k$;

E. APRIORI PT Algorithm

This algorithm is used to build association rule on huge dataset. This can be implemented quickly but it needs more amount of memory space which limits its performances. The pseudocode for apriori PT is given in table 3.

Table 3. Pseudo Code for APRIORI PT Algorithm

For each character has a string and if there is a child node and that the character as a substance.

1. If the character is does not exist to return false state.
2. If the character is exist to repeat the step 1.
3. Do the above steps to continue, until the end of string is reached.
4. When, the end of string is reached the true state is,
5. If the indicator I = NULL (NotLeaf) for the current node
6. Else state is false
7. Return true_state.
8. Else state is true
9. Return false_state.
10. Procedure of find tree and string to begin
11. If tree = NULL then
12. Return FALSE and begin next
13. Increase the index and tree represent as less than the node next
14. And count <- zero
15. While index ->the Not Leaf and count <- 1 to KeyWord and
16. Index ->the children node has represent pChildren[keyword[count]-'a'] is not equal to NULL do
17. Next<- the index -> the children node is represent by index> pChildren[keyword[count]-'a']
18. Index <- next
19. Count is less than the increment of count 1 (count + 1)
20. End while
21. If next = NULL the
22. Return TRUE
23. Else the data <- next
24. If the data -> the word <> keyword then
25. Return TRUE
26. Else
27. If data -> pChildren[26] -> the word <> keyword then
28. Return true_state
29. Else return NULL
30. End.

IV. EXPERIMENTAL RESULTS

The performance factors used for finding the efficiency of Apriori, Apriori PT and Apriori MR are number of association rules generated and execution time. Different thresholds are applied for analyzing the efficiency. This work is implemented in Tanagra tool. TANAGRA tool is open source software and it is an acceptable open source and user friendly computer code package which helps students and researchers for doing their data mining researches [14][15].

Table 4. Apriori Algorithm for Rule Generation

Window Size	Threshold	1000 Ds	2000 Ds	5000 Ds	10,000 Ds
		Rules			
W1	25,55	190	190	212	70
W2		200	198	220	74
W3		236	236	256	76
W4		268	268	320	326
W5		268	268	320	326
W1	45,55	22	22	30	30
W2		24	24	30	24
W3		30	30	30	24
W4		66	66	84	34
W5		40	40	52	54

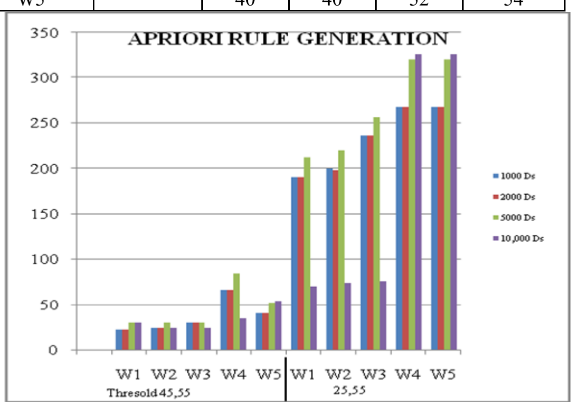

Fig. 2 Rule Generation using Apriori

Table 4. Apriori Algorithm for Time Computation

Window Size	Threshold	1000 Ds	2000 Ds	5000 Ds	10,000 Ds
		Time(ms)			
W1	25,55	109	16	46	78
W2		109	63	63	110
W3		63	32	78	46
W4		62	94	125	125
W5		78	78	63	94
W1	45,55	172	32	46	125
W2		140	63	78	46
W3		109	78	109	110
W4		156	94	125	109
W5		125	109	109	218

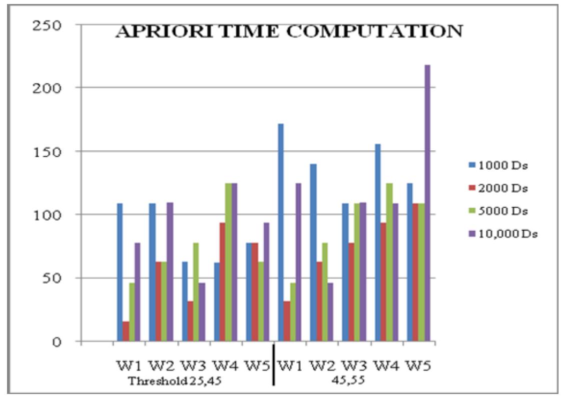

Fig. 3 Apriori Algorithm – Execution Time

Table 6. Apriori MR Algorithm for Rule Generation

Window Size	Threshold	1000 Ds	2000 Ds	5000 Ds	10,000 Ds
		Rules			
W1	25,55	18554	7473	7473	7473
W2		19588	7676	7667	7664
W3		7989	11112	7986	7982
W4		10154	10154	10143	10143
W5		9814	9814	9814	9810
W1	45,55	3000	3090	3112	5553
W2		3003	3060	3045	3078
W3		3689	3652	3691	3634
W4		8465	8434	8490	9456
W5		5497	5493	5461	5449

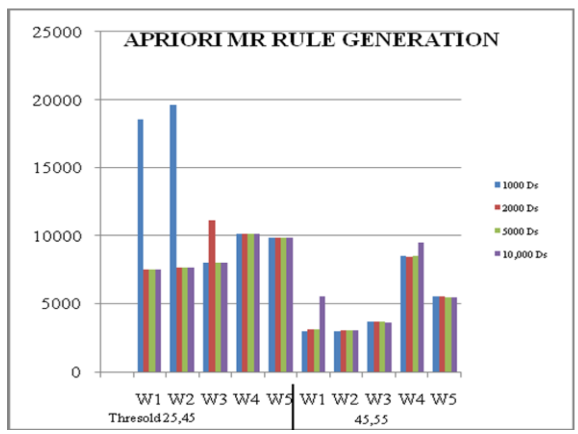

Fig. 4 Rule Generation using Apriori MR

Table 7.Apriori MR Algorithm for Time Computation

Window Size	Threshold	1000 Ds	2000 Ds	5000 Ds	10,000 Ds
		Time(ms)			
W1	25,55	2170	1539	3439	6633
W2		2784	1595	3539	6831
W3		9594	3230	3722	7051
W4		1180	2053	4527	8742
W5		1171	2007	4442	8534
W1	45,55	140	187	296	531
W2		156	203	265	577
W3		203	156	297	515
W4		281	281	609	850
W5		156	219	437	655

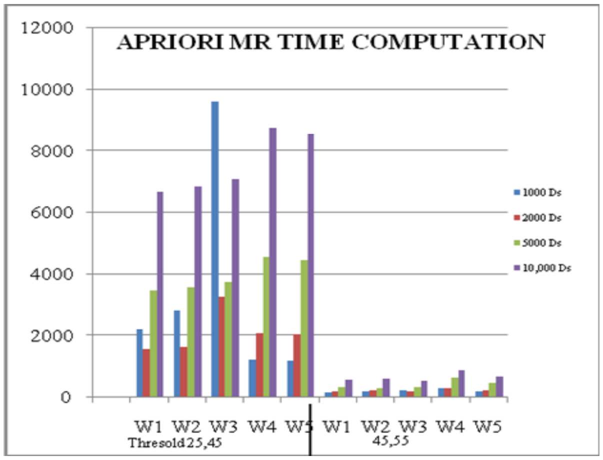

Fig. 5 Apriori MR – Execution Time

Table 8.Apriori PT Algorithm for Rule Generation

Window Size	Threshold	1000 Ds	2000 Ds	5000 Ds	10,000 Ds
		Ruless			
W1	25,55	438	438	438	438
W2		441	441	441	784
W3		458	458	458	809
W4		580	580	580	580
W5		564	564	564	564
W1	45,55	27	27	27	27
W2		28	28	26	27
W3		30	30	30	30
W4		56	56	56	56
W5		42	42	42	42

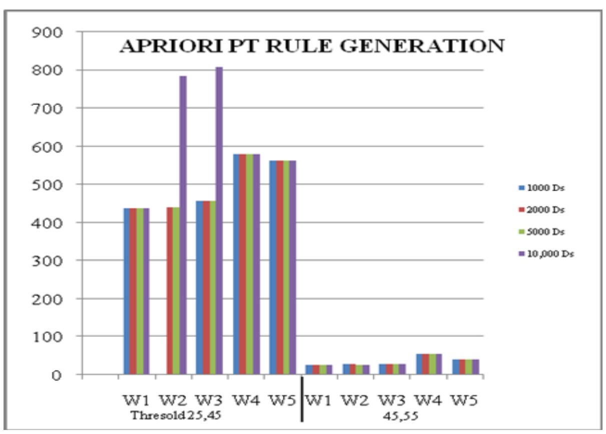

Figure 6. Rule Generation using Apriori PT

Table 9. Apriori PT Algorithm for Time Computation

Window Size	Threshold	1000 Ds	2000 Ds	5000 Ds	10,000 Ds
		Time(ms)			
W1	25,55	0.11	0.11	0.08	0.08
W2		0.12	0.12	0.08	0.08
W3		0.09	0.09	0.08	0.10
W4		0.02	0.02	0.08	0.11
W5		0.09	0.09	0.08	0.06
W1	45,55	0	0	0	0
W2		0.01	0.01	0.01	0
W3		0.02	0.02	0	0
W4		0	0	0	0
W5		0	0	0	0

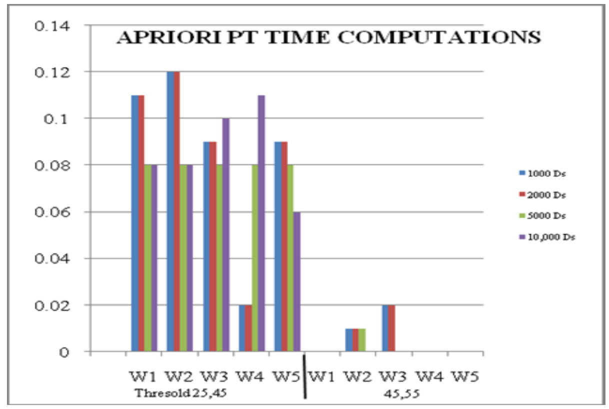

Fig. 5 Apriori PT – Execution Time

CONCLUSION

This paper analyzed different types of apriori algorithms to find the best algorithm for generating association rules. By analyzing the experimental results, we come to know that the performance of Apriori Map/Reduce Algorithmis better than Apriori and Apriori PT. This algorithm generates more number of rules and time computation is very less. This work highlights the data stream association rule generation by providing different support and confidence values and this is applied to different windows.

REFERENCES

[1] Aggarwal C (2003). A Framework for Diagnosing Changes in Evolving Data Streams. ACM SIGMOD Conference.
[2] Agrawal, R. and Srikant, R. Fast Algorithms for Mining Association rules. Proc. 20th VLDB conference, Santiago, Chile, 1994.
[3] Charu C. Aggarwal "Data Stream Models and algorithms"-Data streaming book 2009, Springer.
[4] Christian Hidber. Online Association rule mining. SIGMOD '99 Philadelphia PA. ACM 1-58113-084-8/99/05, 1999.
[5] Charanjeet Kaur, Association Rule Mining using Apriori Algorithm: A Survey ISSN: 2278 – 1323 International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 2, Issue 6, June 2013.
[6] "Data mining techniques "by Arun k Pujari.
[7] "Data Streams: An Overview and Scientific Applications" Charu C. Aggarwal.
[8] "Data Mining: Introductory and Advanced Topics" Margaret H. Dunham.
[9] Frequent item set mining data set repository, http:// fimi.cshelsinki.fi/data/
[10] Han, J., Kamber, M.: "Data Mining Concepts and Techniques", Morgan Kaufmann Publishers, 2006.
[11] Rakesh Agrawal, Ramakrishnan Srikant; Fast Algorithms for Mining Association Rules; Int'l Conf. on Very Large Databases; September 1994.
[12] S.Vijayaraniet al, " Mining Frequent Item Sets over Data Streams using Éclat Algorithm" , International Conference on Research Trends in Computer Technologies (ICRTCT - 2013) Proceedings published in International Journal of Computer Applications® (IJCA) (0975 – 8887) 27.
[13] Nan Jiang and Le Gruenwald, "Research Issues in Data Stream Association Rule Mining"- SIGMOD Record, Vol. 35, No. 1, Mar. 2006.
[14] Website: Tanagra.software.informer.com.
[15] "Mining frequent patterns across multiple data streams" Jing Guo, Peng Zhang, Jianlong Tan and li Guo, 2011.

Mrs. S.Vijayarani has completed MCA and M.Phil in Computer Science. She is working as Assistant Professor in the School of Computer Science and Engineering, Bharathiar University, Coimbatore. Her fields of research interest are data mining, privacy and security issues and data streams. She has published papers in the international journals and presented research papers in international and national

Ms. R.Prasannalakshmi has completed M.C.A in Computer Applications. She is currently pursuing her M.Phil in Computer Science in the School of Computer Science and Engineering, Bharathiar University, Coimbatore. Her fields of interest are Data Streams in data mining and privacy preserving in Data mining.