ASSIGNMENT 6

AIM: Implement Apriori approach for data mining to organize the data items on a shelf using following table of items purchased in a Mall

OBJECTIVE:

- To understand the basic concept of Apriori algorithm.
- To implement Apriori Algorithm.

SOFTWARE REQUIREMENTS:

- Linux Operating System
- Weka Tool

MATHEMATICAL MODEL:
Consider a set S consisting of all the elements related to a program.
The Mathematical model is given as below,
S=s,e,X,Y,Fme,DD,NDD,Mem shared
Where,
s = Initial State
e = End State
X = Input. Here it is number of elements, actual element
Y = Output. Here output is frequent patterns
Fme = Algorithm/Function used in program. for eg. fcal mean(),cal di()g
DD=Deterministic Data
NDD=Non deterministic Data
Mem shared=Memory shared by processor.

THEORY:

Apriori is an algorithm for frequent item set mining and association rule learning over transactional databases. It proceeds by identifying the frequent individual items in the database and extending them to larger and larger item sets as long as those item sets appear sufficiently often in the database. The frequent item sets determined by Apriori can be used to determine association rules which highlight general trends in the database: this has applications in domains such as market basket analysis.

The Apriori algorithm was proposed by Agarwal and Srikant in 1994. Apriori is designed to operate on databases containing transactions (for example, collections of items bought by customers, or details of a website frequentation). Other algorithms are designed for finding association rules in data having no transactions, or having no timestamps (DNA sequencing). Each transaction is seen as a set of items (an item set). Given a threshold, the Apriori algorithm identifies the item sets which are subsets of at least transactions in the database.

Apriori uses a "bottom up" approach, where frequent subsets are extended one item at a time (a step known as candidate generation), and groups of candidates are tested against the data. The algorithm terminates when no further successful extensions are found.

Apriori uses breadth-first search and a Hash tree structure to count candidate item sets efficiently. It generates candidate item sets of length from item sets of length . Then it prunes the candidates which have an infrequent sub pattern. According to the downward closure lemma, the candidate set contains all frequent -length item sets. After that, it scans the transaction database to determine frequent item sets among the candidates.

- The pseudo code for the algorithm is given below for a transaction database, and a support threshold of. Usual set theoretic notation is employed; though note that is a musti set. is the candidate set for level . At each step, the algorithm is assumed to generate the candidate sets from the large item sets of the preceding level, heeding the downward closure lemma. Accesses a field of the data structure that represents candidate set , which is initially assumed to be zero. Many details are omitted below, usually the most important part of the implementation is the data structure used for storing the candidate sets, and counting their frequencies.

Example No.	Color	Type	Origin	Stolen?
1	Red	Sports	Domestic	Yes
2	Red	Sports	Domestic	No
3	Red	Sports	Domestic	Yes
4	Yellow	Sports	Domestic	No
5	Yellow	Sports	Imported	Yes
6	Yellow	SUV	Imported	No
7	Yellow	SUV	Imported	Yes
8	Yellow	SUV	Domestic	No
9	Red	SUV	Imported	No
10	Red	Sports	Imported	Yes

Items	Count
Mango	3
Onion	3
Jar	2
Keychain	5
Eggs	4
Chocolates	3
Apple	1
Corn	2
Knife	1
Nut	1

Table: L1

C1

I1	Mango	3
I2	Onion	3
I3	Jar	2
I4	Key-chain	5
I5	Eggs	4
I6	Chocolates	3

L2

I1,I2	1
I1,I3	1
I1,I4	3
I1,I5	2
I1,I6	1
I2,I3	2
I2,I4	4
I2,I5	4
I2,I6	2
I3,I4	2
I3,I5	2
I3,I6	2
I4,I5	4
I4,I6	3
I5,I6	2

C2

I1,I4	3
I2,I4	4
I2,I5	4
I4,I5	4
I4,I6	3

I1,I4,I5	2
I1,I4,I6	3
I2,I4,I5	4
I2,I4,I6	2

C3

I1,I4,I5	3
I2,I4,I5	4

Hence these are the item sets which occurred frequently in the database .

CONCLUSION: Thus, we have implemented Apriori Algorithm.

Roll No.	**Name of Student**	**Date of Performance**	**Date of Submission**	**Sign.**
BECOC357	Sunny Shah	21 / 09 / 2017	28 / 09 / 2017	