DATE:

Assignment No : DMW-3

* Title : Apriori Algorithm

* Problem statement :

Apply Apriori algorithm to find frequently occuring items from given data and generate strong association rules using support and confident thresholds eg: Market Basket Analysis.

* Objectives : To,

Apply apriori algorithm for market basket analysis and operate strong association rules.

* Outcomes : Students will be able to,

Generate frequent itemset of transactions such that support value is greater than minimum support.

* S/w and H/w requirements :

Linux /windows 10 , inten i5 processor , 64-bit PC , 500 GB HDD.

* Concepts related to theory :

Market Basket Analysis association and correlation between transactions and frequent itemsets.

Association notes -

$A \Rightarrow B$ [support = x% confidence]

DATE:

A transaction T contains set of item A if $A \subset T$

$A \Rightarrow B$ where $A \subset I$, $B \subset I$ (I = Itemset)

$A \neq \phi$, $B = \phi$ & $A \cap B = \phi$

Support 's' = % of transaction in dataset 'D' containing $(A \cup B)$

confidence 'c' = % of transactions in 'D' that containing A & B.

$s(A \Rightarrow B) = P(A \cup B)$

$c(A \Rightarrow B) = P(B/A) = \frac{\text{support}(A \cup B)}{\text{support}(A)}$

Strong Association - when support and confidence is greater than min_support and min_confidence

Apriori property -

All non-empty subsets of a frequent itemset must also be frequent

Generate association rules -

- For each frequent itemset l, generate all non-empty subsets of l.
- For every non-empty subset s of l

output : $s \Rightarrow l-s$ if $\frac{\text{support}(l)}{\text{support}(s)} \geq$ min_support

* Algorithms -

1) Calculate the support of items sets of size k=1 in the transactional ~~bater~~ databases. This is called generating the candidate set.

DATE:

2) Prune the candidate set by eliminating items with a support less than the green threshold.

3) Join the frequent itemset (of size K+1) and repeat the above sets until no more itemsets can be informed. This will happen when the sets formed have support less than given support.

eg: support = 3, confidence = 80%.

Transaction database:

Transaction ID	Items
T_1	I_1, I_2, I_3, I_4
T_2	I_2, I_3
T_3	I_3, I_4
T_4	I_2, I_3, I_4

For rule $X \rightarrow Y$

support $(x \& y)$ / support (y)

The following can be obtained from the size of two frequent itemsets.

1) $I_2 \rightarrow I_3$ confidence = 3/3 = 100%.
2) $I_3 \rightarrow I_2$ confidence = 3/4 = 75%.
3) $I_3 \rightarrow I_4$ confidence = 3/4 = 75%.
4) $I_1 \rightarrow I_3$ confidence = 3/3 = 100%.

Since our required confidence is 80%, only rule 1 & 4 are included in the results.

Therfore, it can be concluded that customers who bought item (I_2) always bought item (I_3) with it. Also the customer who bought (I_4) always bought item (I_3) with it.

DATE:

Applications -
1) Education field
2) Medical field
3) Business optimization etc.

Advantages:
1) It is easy to understand and implement
2) Join and prune algorithm are easy to process on large itemsets

Disadvantages:
1) Requires high computation, if data of itemsets is very large and minimum support is kept very low.
2) The entire database needs to be scanned.

* Conclusion:
We have successfully implemented and learned Apriori algorithm.