Data Mining Using Apriori And FP-Tree Algorithm

Dr. A.Carmel Prabha, Technical Test Lead, Infosys Limited, Chennai.

Article History
Received: 22.04.2022
Revised and Accepted : 30.04.2022
Published: 25.06.2022

https://doi.org/10.56343/STET.116.015.004.007
www.stetjournals.com

Abstract

This document evaluates two important data mining techniques i.e., Apriori Algorithm and FP Tree technique to sort data from the data warehouse.

Keywords: Apriori Algorithm, Automated Data Analysis, Data Mining, Data Warehousing, FP Tree Technique

INTRODUCTION

An automated analysis of massive data sets in a Data Mart is known as Data Mining. Moreover, it has the definition of finding hidden information within a database. Various algorithms and techniques are used in Data Mining to accomplish different tasks (Ramakrishnan and Gehrke, 2003; Bishop, 2006; Ian *et al*., 2011). Based on the characteristics of the data being examined, these algorithms determine and fit the model that fits the data the closest.

There are two types of data mining tasks:

Predictive: In this type of analysis, predictions are made based on previous results.

Descriptive: The data is described in a descriptive manner by identifying patterns or relationships.

1.Data Mining Versus Data Warehousing

Statistical methods and clustering techniques are used in data mining to find hidden patterns and trends in operational data.The data mining process summarizes data and can be used by data warehouses to provide business intelligence with faster processing times.

Using data mines for analysis of the data in a data warehouse can speed up the processing of the data.

Dr. A.Carmel Prabha
email : prabha200481@yahoo.co.in
Technical Test Lead, Infosys Limited, Chennai.

2.Some Types of Data Mining Techniques:

There are many data mining techniques available to sort out required data from data warehouse. Out of those two commonly known techniques are:-

- Apriori Algorithm
- FP-Tree

Apriori Principle: -

The Apriori principle states that:" If an itemset is regular then all of its subcategories must also be common". This involves repeatedly refining the item sets to retain only frequent item sets and discard other.

Itemset: Items in transaction

Frequent pattern and item Sets: -

In a data warehouse some data patterns appear more regularly than the remaining data such a data items are called as frequent patterns or frequent item sets

Association Rule:-

Denoted as X&Y somewhere X besides Y be situatedgroups of substances, transactions that contains items in X generally also contains items in Y

The power of an affiliation rule is measured the use of two phrases:-

Confidence: How regularly gadgets in Y show up in transaction that additionally includes X.
Confidence tells us how reliable the conclusion of rule is.

Support: Just howrepeatedly a regulation is valid to certainnumbersestablished.

Support tells us in what way often a rule is likely to be existing in practice, so low support means the business would not be interested in that rule.

P - ISSN 0973 - 9157
E - ISSN 2393 - 9249

Calculation of support and confidence:

Support is calculated primarily based on help remember and complete wide variety of transactions, here support count means the frequency of the rule that we are trying to test.

For example: If our rule is {milk, biscuits} —> {fruits}

Then support count will be number of transactions that have all this three items.

Example:

Consider following transaction

{bread, milk}

{bread, biscuits, fruits, eggs}

{milk, biscuits, fruits, cola}

{bread, milk, biscuits, fruits}

{bread, milk, biscuits, cola}

Now consider the rule

{Milk, biscuits} —> {fruits}

We have assist relycount for {milk, biscuits, fruits} as two and complete wide varietyof transactions are 5.

Hence the aid for the rule=2/5=0.4

Rule's confidence=support count for {milk, biscuits, fruits}/ support count for {milk, biscuits} =2/3=0.7

Example

ALL ELECTRONICS TRANSCATION DATABASE D: (|D|=9)

Trans. ID	List OfObject _Id's
Trans100	O1, O2,O5
Trans200	O2,O4
Trans300	O2,O3
Trans400	O1,O2,O4
Trans500	O1,O3
Trans600	O2,O3
Trans700	O1, O3
Trans800	O1, O2, O3, O5
Trans900	O1, O2, O3

Apriori Algorithm for outcomerecurrent itemsets in D.

1. Popular the primarygroup of the procedure, every object is a participant of the conventional of applicant l-item groups, C1. The proceduredefinitely examinationsaltogether of the connections to be counted the wide variety of incidences of every entry.
2. Assume that the minimal help sum number essential is two, that is min_sup=2. (The equivalentqualified assist is 2/9=22%). The traditional of accepted l-itemset, L1, containerat that moment be resolute. It contains of the applicant l-itemgroups fulfilling minimal provision.
3. Towards find out the conventional of regular 2-itemgroups, L2, the process makes use of the be part of L1 | X| L1 to produce a candidate set of two itemgroups C2.
4. Subsequent, the connections in D areperused and the assist depend of every applicant itemtraditional in C2 is amassed, as exposed in the below board for C2.
5. Follow similar steps while obtaining C3 where we discover frequent 3-itemset that

   is L3.

Solution:

Examinationthe ObjectD for sum of to each applicant:

C1:

Elementset	Element.count
{E1)	6
{E2}	7
{E3}	6
{E4}	2
{E5}	2

Comparison applicant aid sum by least aid sum:

L1:

Elementset	Each.count
{E1}	6
{E2}	7
{E3}	6
{E4}	2
{E5}	2

Create C2 candidate on or after L1

Element Group
{E1, E2}
{E1, E3}
{E1,E4}
{E1, E5}
{E2, E3}
{E2, E4}
{E2, E5}
{E3, E4}
{E3, E5}
{E4, E5}

Examination of the Object D designed for sum of to each applicant

C2:

Element Group	Each.count
{E1, E2}	4
{E1, E3}	4
{E1, E4}	1
{E1, E5}	2
{E2, E3}	4
{E2, E4}	2
{E2, E5}	2
{E3, E4}	0
{E3, E5}	1
{E4, E5}	0

Comparison applicant provision quantity through smallest provision amount

L2:

Element Group	Each.total
{E1, E2}	4
{E1, E3}	4
{E1, E5}	2
{E2, E3}	4
{E2, E4}	2
{E2, E5}	2

Create C3 applicants from L2

C3:

Element Group
E1, E2, E3}
{E1, E2, E5}

Examination of the Object D designed designed fors um of to each applicant

C3:

Element Group	Each.count
{E1, E2, E3}	2
{E1, E2, E5}	2

Comparability applicant aidsum with least aid sum

L3:

Element Group	Each.count
{E1, E2, E3}	2
{E1, E2, E5}	2

FP-Tree:-

Apriori algorithm is called as generate ant test approach. If we expand this logic and represent the frequent items sets in a tree like graphical structure then it is called as FP-Tree.

FP-Tree is flattened, graphical symbol of the involvementfigures. To construct an FP-Tree the transactions from an object set are examine one at a time and mapped onto a course in a tree.Since distinct transactions can have frequent gadgets there branches will overlap, more the overlap more is the compression of the tree.

Example:-

Consider the following frequent item sets or transactions in a database, find the items with minimum +support as 30% and create an FP-Tree.

Trans. ID	**Pieces**
Trans1	E,A,D,B
Trans 2	D,A,C,E,B
Trans 3	C,A,B,E
Trans 4	B,A,D
Trans 5	D
Trans 6	D,B
Trans 7	A,D,E
Trans 8	B,C

Solution:-

Step1: Calculate leastaid count = (minimum support/100)*(total number of transactions)

= 30/100*8=2.4 (we will round it to 3)

Step2: Now find the frequency for each item, and decide priority based on frequency that means, Give priority as 1 to the item with highest frequency then 2 for next highest and if frequencies are same for items then priority order can be taken in the way we desire means any one can put first

Piece	**Occurrence**	**Precedence**
A	5	3
B	6	1
C	3	5
D	6	2
E	4	4

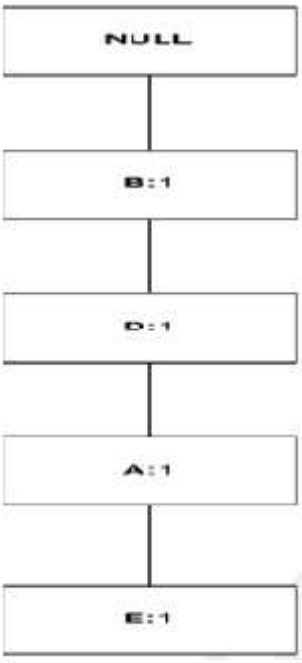
Step 3: Drop the objects whose frequency is less than 3

Step 4: Direction the objectsconferring to the urgencies

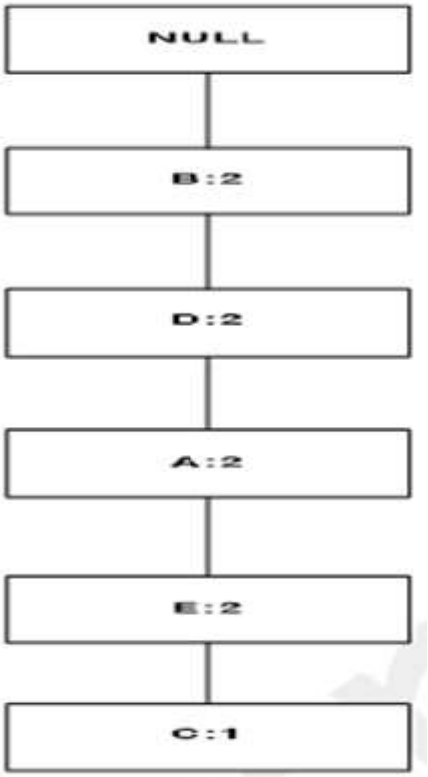
Trans.ID	Piece	Ordered Piece
Trans1	E,A,D,B	B,D,A,E
Trans2	D,A,C,E,B	B,D,A,E,C
Trans3	C,A,B,E	B,A,E,C
Trans4	B,A,D	B,D,A
Trans5	D	D
Trans6	D,B	B,D
Trans7	A,D,E	D,A,E
Trans8	B,C	B,C

Step 5: Enticement FP-Tree, every FP-Tree has 'Null' as the root node, so draw the null root node and attach the items from row one, also write the count or occurrences of items

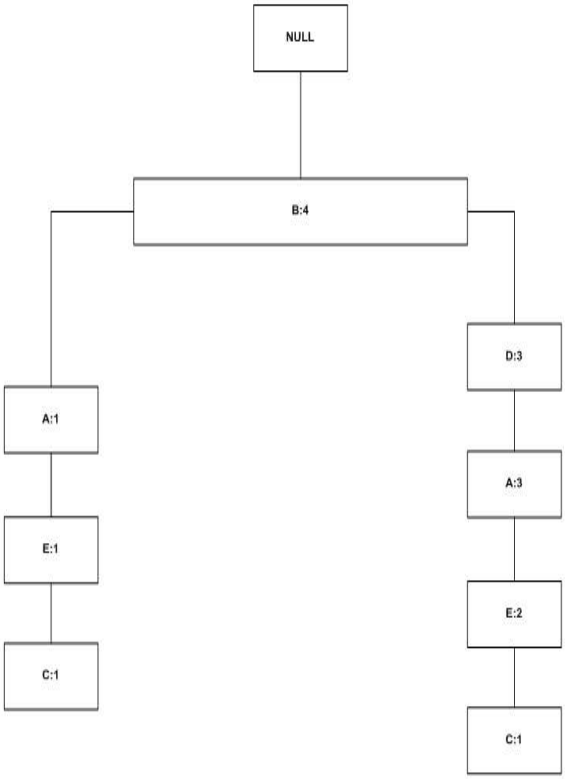
5.1: Read first row and attach items with the count



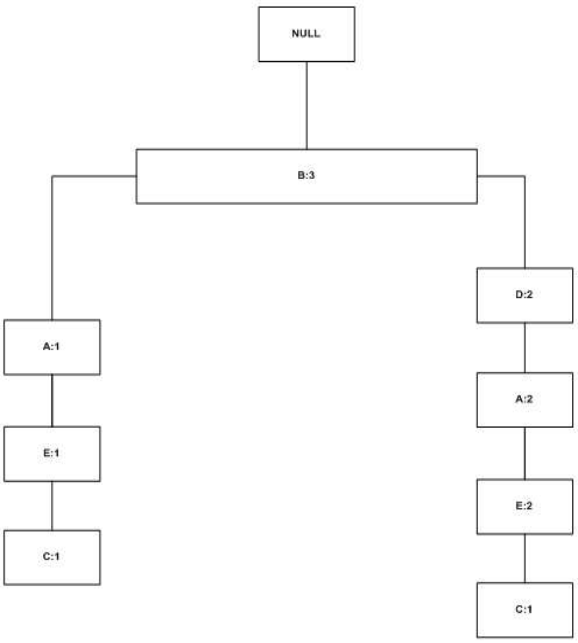
5.2: Read second row and update the tree i.e. if the same node already exists simply increment count and if not then form new branch with the counter initiating from 1 for the Objects of new branch.



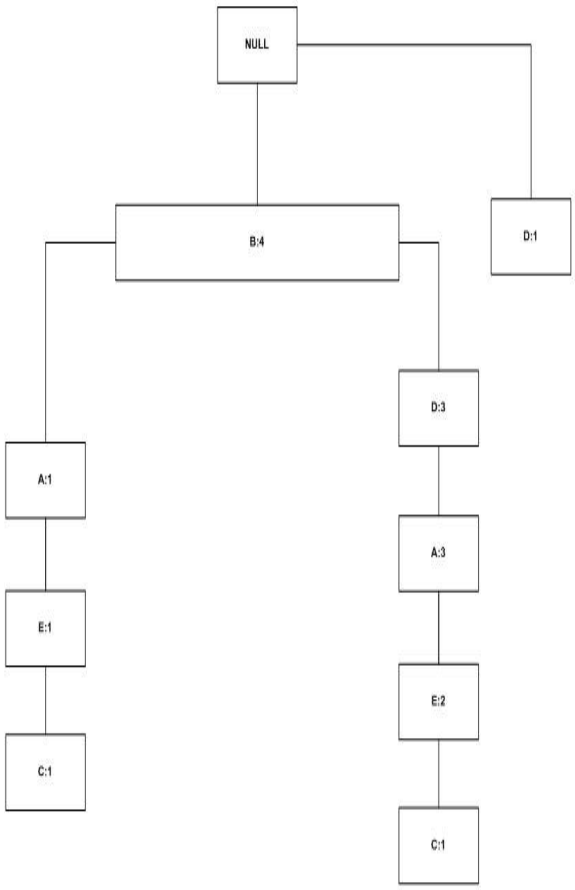
5.3: Read 3rd row, here we were not having B-A-E-C so we attached new branch A-E-C to node B as till node B we were having.



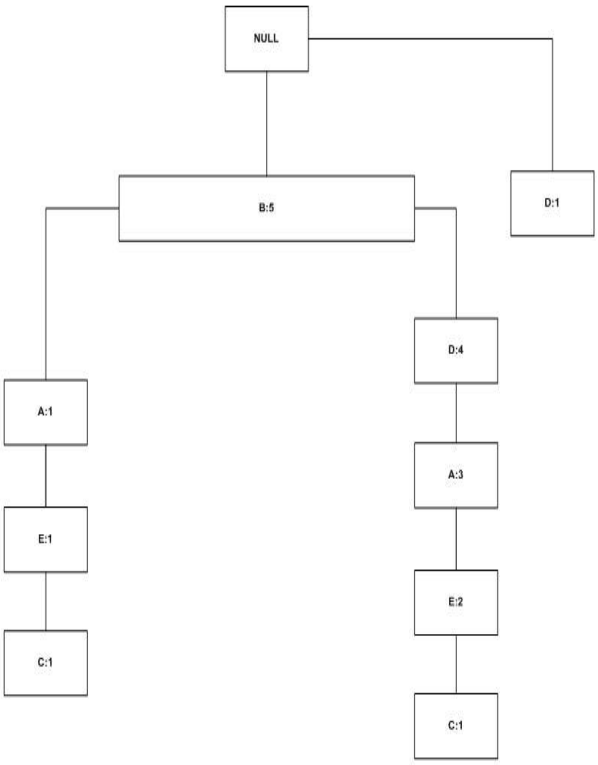
5.5: Read fifth row, here we formed new branch with node as D attached to NULL.



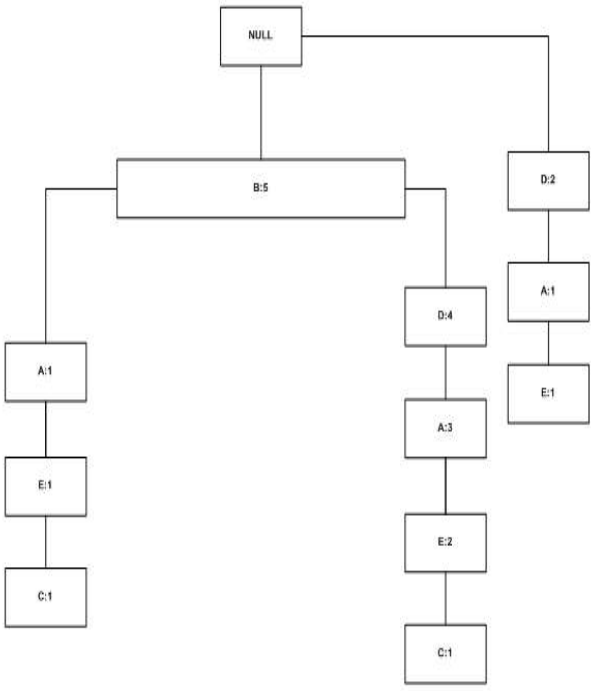
5.4: Read fourth row, here we simply increased count for B-D-A



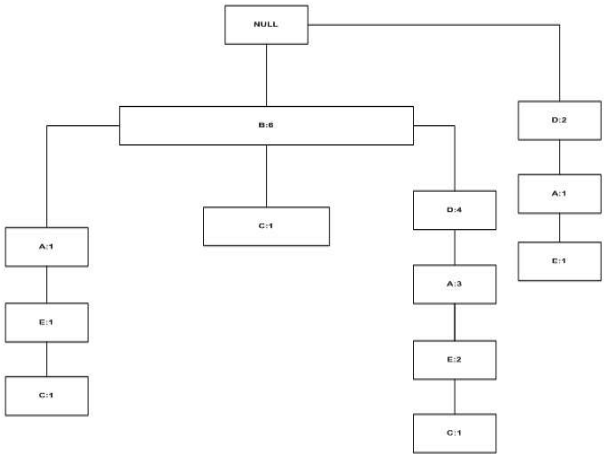
5.6: Read sixth row, here we simply incremented counts of B-D



5.7: Read seventh row, here we formed new branch D-A-E



5.8: Read 8th row, here we have formed new branch B-C.



CONCLUSION

In this way these two techniques can be used to sort out essential data from huge data stored in data warehouse, which helps to get frequently occurring data items. The study of frequently occurring data patterns is very useful in different business firms to know which products customer buys more frequently. These techniques are also useful in various organizations to perform different analysis on the historical data.

References

Bishop, Christopher M. 2006.Pattern Recognition and Machine Learning, Springer, New York, NY.

Ramkrishnan, Raghu and Gehrke, Johannes.2003.Database ManagementSystems, ,Second Edition, McGraw Hill International: Computer Science Series, New York.

Ian H.Witten, Eibe Frank and Mark A. Hall. 2011. Data Mining: Practical Machine Learning Tools and Techniques, Elsevier/(Morgan Kauffman), ISBN:97893 80501864. New York.