International Journal of Emerging Technologies in Engineering Research (IJETER)
Volume 4, Issue 5, May (2016) www.ijeter.everscience.org

An Approach For Mining Association Rules For 3d Data Using Representative Slice Mining (Rsm) Framework

R Komala
Assistant Professor
Computer Science and Engineering
JNTUK-UCEV
Vizianagaram, Andhra Pradesh, India

P. R. S. Naidu
Assistant Professor
Computer Science and Engineering
JNTUK-UCEV
Vizianagaram, Andhra Pradesh, India

N. K. Sumanth
Assistant Professor
Computer Science and Engineering
JNTUK-UCEV
Vizianagaram, Andhra Pradesh, India

Abstract— In the present trends of data mining, the Mining frequent patterns are significantly important. Over the past few decades many of the efficient FCP mining algorithms have been in the literature which includes feature enumeration algorithms, row enumeration algorithms and dense data mining algorithms. In addition, there is a limitation on all these algorithms to 2D dataset analysis. Some of the 3D application areas are gene-sample-time microarray data, transaction-item-location market-basket data. The existing data mining algorithms like CLOSET, CHARM and D-Miner are used to extract the Frequent Closed Cubes (FCC) from a 3D dataset. These algorithms endeavor to mine Frequent Closed Cubes that give "close" relationships among three dimensions. There is no possibility in furtherance of expansion in any dimension can be made on the pattern.

Representative Slice Mining (RSM) is a three phase framework, which makes use of existing 2D FCP mining algorithms to mine 3D FCCs. In phase 1, representative slice is developed based on one dimensional classification and slices combination. In phase 2, to mine 2D Frequent Closed Patterns a 2D frequent closed pattern mining algorithm can be applied on each representative slice. In phase 3, a post-pruning method is implicated to remove Frequent Closed Cubes unclosed in the classified dimension.

Extension to the existing system is generation of Association Rule Mining which can be further used in classification. Association Rule Mining is used in many application domains for finding interesting patterns. One of the best known application areas is the market-basket analysis where purchase patterns are discovered and further association analysis is useful for decision making and effective marketing.

Index Terms - component; formatting; style; styling; insert (key words).

1. INTRODUCTION

The process of Association rule mining is defined as extracting desired relations between attributes in huge databases. Using various measures of interestingness the strong rules are discovered in large databases. Besides, the association rule mining is defined as a given set of transactions in finding rules that will predict the occurrence of an item based on the occurrences of other items in the transaction.

Data Mining: Data mining is the automatic acquisition of patterns that represent the knowledge implicitly stored in the data. Association rule mining is one form of data mining that discovers associations among attributes or items of transactions. There are two stages included in the association rule mining process: the pattern mining stage and the rule generation stage. The former is the process used to obtain frequent patterns, where a pattern is a set of items or attributes, and is considered a frequent pattern if it occurs in many transactions. From the discovered frequent patterns the association rules are generated.

Association Analysis: Several business corporate enterprises procure data in large amounts from their regular operations. A simple illustration is collection of customer purchase data in huge amounts at paying counters of shopping malls. Table 1.1 describes a market basket analysis. Each row in this table indicates a transaction having a unique identifier labeled TID along with set of items. By analyzing the data, retailers will conclude the purchasing behavior of their customers. The analyzed data is consider as an important information which is further assist a range of business applications matching inventory management, promotions on marketing, and customer relationships.

TID	Items
1	{Britania,Ghee}
2	{Britania,Goodday,Thumsup,Eggs}
3	{Ghee,Goodday,Thumsup,Pepsi}
4	{Britania,Ghee,Goodday,Thumsup}
5	{Britania,Ghee,Goodday,Pepsi}

Table 1.1: An Example of Market Basket Transactions

Frequent Patterns: Especially in data mining an item set is a confined group or set of elements or items which are represented together as a single entity; absolutely it's a pattern type and often occurs frequently. For example consider data mining project in an insurance company which deals with life insurance instruments. The main motto is not considering what types of policies they sell. Now analyze the additional privileges which are included. These additional privileges referred as riders. The additional benefits (riders) that an insurance company offers are listed below:

G1 Waiver of premium for disability benefit

G2 Disability income benefit

G3 Dismemberment benefit

G4 Accidental death benefit

G5 Waiver of premium for payer benefit

G6 Terminal illness benefit

G7 Dread disease benefit

G8 Long term care

G9 Wife/Husband and kid's Insurance benefit

G10 Children's Insurance Rider

G11 Second Insured Rider

G12 Guaranteed insurability benefit

G13 Paid-up additions option benefit

Closed Frequent Patterns:

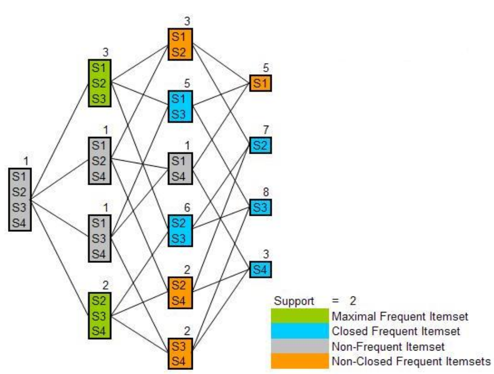


Fig 1.1: Frequent Patterns

The above Fig 1.1 describes, the individual items S1, S2, S3, S4 are frequent item sets as value of support is more than two. Apart three are closed because S1 superset (S1, S3) claiming same support. Below item sets (S1, S2, S3) and (S2, S3, S4) carries frequent property this is due to their presence in minimum two of them.

3D Data: Three-Dimensional (3D) data, is a predominant process of data gathering in recent years,3D datasets have been used for gathering process in Molecular biology (gene-sample-time microarray data), finance management(stock-financial ratio-year data), market analysis (item-time-region data). Fig 1.2 represents a 3D dataset as a cuboid D = O × A × T, where O be a set of objects, A be a set of attributes, T be timestamps.

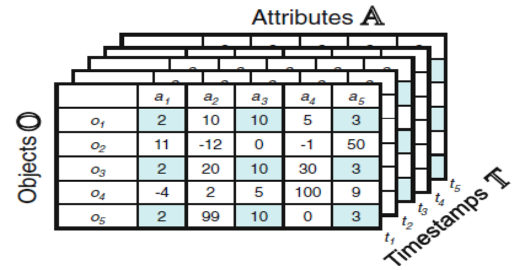


Fig 1.2: A 3D dataset as a cuboid D=O x A x T

Closed Cube: Data cubes are used for data storage by Data warehouses which are multi-dimensional databases. Each data cube consist a set of well-defined dimensions, with each dimension organized in hierarchy levels. Moreover a data cube measures which are aggregations based on the dimension hierarchies.

A closed cube represents a size-reduced representation of a data, it only consists of a closed cells. A cell, say c, is a closed cell if there is no cell, d, such that d is a specialization (descendent) of c, and d has the same measure values as c.

Frequent Closed Cube: A Frequent closed cube is defined as the closed cube which satisfies the minimum support. Finding these closed frequent cubes can be of a great help to get a lot of cubes that are not necessary and to find the right associations rules.

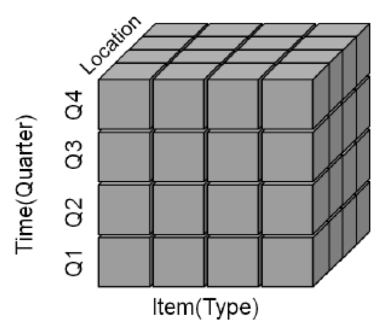


Fig 1.3: 3D frequent closed cube

2. LITERATURE SURVEY

Some efficient FCP mining algorithms are feature enumeration algorithms, row enumeration algorithms and dens data mining algorithms. The examples of Feature enumeration algorithms are CHARM and CLOSET. These are efficient for datasets with small number of "features" and large number of rows. Row enumeration algorithms like CARPENTER, algorithm is more suitable for datasets with large number of features and small number of rows. Dense data mining algorithms like D-Miner are more suitable for datasets which are highly dense and correlated.

2.1 Feature Enumeration Algorithms

These algorithms include CHARM and CLOSET.

2.1.1 CHARM (Closed Hierarchical Association Rule Mining)

All previous association mining methods used to just exploit the item set search space. But CHARM, on the other hand, is unique unlike them. It explores both the item set space and transaction space together at same time. In addition to this, CHARM, while calculating the closed frequent sets, avoids calculating all possible subsets of a closed item set. The inspection of both the item set and transaction space enables CHARM to use a innovative search approach that skips many levels to promptly determine the closed frequent item sets, rather than having to calculate many non-closed subsets. Further, CHARM uses a two-way elimination procedure. It excludes candidates depending not only on subset infrequency (i.e., not even one extension of an infrequent are tested) as all the other association mining methods do, but it also excludes candidates based on non-closure property, (i.e., any non-closed item set is eliminated). Lastly, CHARM does not uses internal data structures like Tries or Hash-trees. The elementary procedure used is a combination of two item sets and an intersection of two transactions lists where the item sets are accommodated. An broad set of experiments certifies that CHARM gives orders of magnitude enhancement over current processes for mining closed item sets, even over processes like AClose, that are especially developed to mine closed item sets. It makes lesser database scans than the longest closed frequent set found, and it makes similar number of transactions and also finds similar number of closed item sets.

2.1.2 CLOSET (CLOSed sET)

CLOSET, for mining closed item sets, with the development of three procedures:

- Employing a compressed, frequent pattern tree FP-tree structure for mining closed item sets without candidate set formation,
- Building up a single prefix path compression technique to identify frequent closed item sets quickly, and
- Probing a partition-based projection system for extensible mining in huge databases.

2.2 Row Enumeration Algorithms

2.2.1 CARPENTER

CARPENTER (Closed Pattern Discovery by Transposing Tables that are extremely long; the “ar” in the name is gratuitous.) is specifically composed to deal with datasets having a large number of attributes and comparatively lesser number of rows. Many experiments on original bioinformatics datasets depict that CARPENTER is orders of magnitude better than earlier closed pattern mining algorithms like CLOSET and CHARM. CARPENTER is a unique algorithm which finds frequent closed patterns by doing depth-first row-wise calculation rather than the conventional item set calculation, mixed with effective search pruning methods, to generate a great optimized algorithm.

2.3 Dense Data Mining Algorithm

2.3.1 D-Miner

One of the most well-known data mining methods concerns transactional data analysis by making use of set patterns. Transactional data can be presented as boolean matrices (see Figure 2.1). Lines stand for transactions and columns represent boolean attributes that permit record item occurrences. For example, in Figure 2.1, transaction t4 contains the items g5, g6, g7, g8, g9, and g10. The frequent set mining issue worries the calculation of sets of attributes that are true together in sufficient transactions, i.e., at a given frequency threshold.

	Items									
	g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8	g_9	g_{10}
t_1	1	1	1	1	0	1	1	0	0	0
t_2	1	1	1	1	0	0	0	0	1	1
t_3	1	1	1	1	0	0	0	0	1	1
t_4	0	0	0	0	1	1	1	1	1	1
t_5	1	0	1	0	1	1	1	1	0	0

Table 2.1: Example of Boolean Context

The regular case of basket analysis (large – ultimately millions – number of transactions, hundreds of attributes, but inadequate and less correlated data) can be managed by various algorithms, along with the different Apriori-like algorithms that have been developed over the last decade. These algorithms fail if the data are dense and highly-correlated, but the purported condensed depictions of the frequent itemsets can be calculated. For example, competent algorithms can calculate the frequent closed sets from which, without using the data, every frequent set and its frequency can be computed. Remaining significant applications concern datasets with only a few transactions, e.g., for usual gene expression data where items represent gene expression properties in biological circumstances. Anyhow, it is feasible to utilize the properties of Galois connection to calculate the closed sets on the smaller dimension and determine the closed sets on the other dimension. In this case, we consider bi-set mining in challenging cases, i.e., when the data is dense and when not even one of the dimensions is very less. Bi-sets consists of a set of lines T and a set of columns G. T and G can be correlated by many relationships, e.g., all the items of

G are a part of each transaction of T (1-rectangles). It is amusing to restrain more bi-set components to be closed sets (likewise called maximal 1-rectangles or concepts). Other constraints like minimal and maximal frequency can also be used. We offer an original algorithm called D-Miner which calculates concepts under constraints. It works in another way from various other concept discovery algorithms and (frequent) closed set calculation algorithms. D-Miner can be employed when the earlier algorithms usually fail in dense Boolean datasets. With the active use of the constraints, it enhances the applicability of concept discovery for matrices whose dimensions are very large.

3. EXISTING SYSTEM

- To tackle the problem of mining FCC from 3D datasets Representative Slice Mining has been utilized. Our contributions are as follows. The first approach is a three-phase framework that exploits 2D FCP mining algorithms to mine FCCs. The basic idea is to transform a 3D dataset into a set of 2D datasets, mine the 2D datasets using an existing 2D FCP mining algorithm, and then prune away any frequent cubes that are not closed.
- In this paper, we attempt to mine FCCs that deliver "close" relationships among three dimensions. That is, we want to find the maximum patterns in a 3D context. The 3D pattern is maximum in that an increase in any dimension will cause a direct decrease in at least one of the other two dimensions; i.e., no further expansion in any dimension can be made on the pattern.

4. ABOUT RSM (REPRESENTATIVE SLICE MINING)

RSM includes three phases. They are

1. Representative Slice Generation
2. 2D FCP Generation
3. 3D FCC Generation by Post-pruning

Explanation:

Example of Binary Data context

P/Q	Q1	Q2	Q3	Q4	Q5
P1	1	1	1	0	1
P2	1	1	1	0	0
P3	1	1	1	1	
P4	0	0	1	0	1

R=R1

P/Q	Q1	Q2	Q3	Q4	Q5
P1	1	1	1	1	1
P2	0	1	1	1	0
P3	1	1	1	1	0
P4	1	1	1	0	1

R=R2

P/Q	Q1	Q2	Q3	Q4	Q5
P1	1	1	1	0	0
P2	1	1	1	0	0
P3	1	1	1	1	0
P4	1	1	0	1	1

R=R3

1. Representative Slice Generation

In 1st phase, we first consider the height dimension H as our base dimension1, and calculate set R = {R1, R2, . . . , Rl} to derive all subsets of R(denoted R˙) so that |R˙| >= minR.

The dataset is shown in above Tables for instance, let's assume minR = 2, then we will obtain the subsets {R1, R2}, {R1, R2, R3}, {R1, R3}, and {R2, R3}. Second, slices that are in the same subset are joined to build a new representative slice (RS). For a 3D dataset O = R × P × Q = $\{O_{k,i,j}\}$ with k ∈ [1, l], i ∈ [1, n] and j ∈ [1,m], and let R˙ = {R1, . . . , Rx} be the subset that has to be joined. Then the RS of R˙ can be depicted as a n × m matrix so that $\forall O'_{i,j} \in RS, O'_{i,j}$ where i ∈ [1, n] and j ∈ [1,m], i.e., only when all of its make-up values are 1, the cell value of the representative slice is 1. Else, the cell value is 0. Then it can be stated that the heights in R˙ "contribute to" the RS of R˙.

Height Sets	Representative Slice
R1,R2	1 1 1 0 1 0 1 1 0 0 1 1 1 1 0 0 0 1 0 1

Table 3.2: Representative Slice Generation

2. 2D FCP Generation

In 2nd phase, any current FCP mining algorithm can be used on each representative slice to mine 2D FCPs based on dimensions R and C. In our experiments, we used D-Miner as it has been found to be impressive on relatively dense datasets with long patterns. Next to mining, we will have a set of 2D FCPs for R and C dimensions.

D- Miner Principle

D-Miner is a new algorithm for drawing out concepts (*T*,*G*) under constraints. It constructs the sets *T* and *G* and it employs monotonic constraints on *LO* and *LP* at the same time to decrease the search space. A concept (*T*,*G*) states that all its items and objects are in relation by *R*. Since there is no relation between an item *g* and an object *t*, it generates two concepts, one with *g* and the other without *t*, and another one with *t* and without *g*. D-Miner is based on this fact. Lets denote a set of 0-rectangles by *H* such that it is a partition of the false values (0) of the Boolean matrix, i.e., $\forall g \in P$ and $\forall t \in O$ such that $(t, g) \in \mathbf{r}$, it prevail one and only one element (X, Y) of *H* such that $t \in X$ and $g \in Y$ .The elements of *H* are called *cutters*. *H* should be as small as possible to minimize the depth of recursion thereby decreasing the execution time. On the other side, we should not spend too much time to calculate *H*. *H* contains a lot of elements same as lines in the matrix. Each element consists of the attribute valued by 0 in this line. Time complexity for calculating *H* is in $O(n \times m)$ where *n* and *m* are the dimensions of the matrix. Calculating time is insignificant when compared to the one of the cutting procedure. In addition to this, using this definition makes the pruning of 1-rectangles simpler that are not concepts.

D-Miner, after starting initially with the couple (*O*, *P*), splits it recursively until *H* is empty using the elements of *H* and therefore each couple is a 1-rectangle. An element (*a*, *b*) of *H* is used to cut a couple (*X, Y)* if $a \cap X \neq \emptyset$ and $b \cap Y \neq \emptyset$. We can define the left son of (*X, Y*) by ($X \backslash a$, *Y*) and the right son by (X, $Y \backslash b$). Recursive splitting draws to all the concepts, i.e., the maximal 1-rectangles but also some non-maximal ones.

D-Miner Algorithm:

Algorithm 1: **D-Miner**

Input : Database **r** with **n** lines and **m** columns, **O** the set of objects ,**P** the set of items, $\mathbf{C_t}$ and $\mathbf{C_g}$ are monotonic constraints on **O** and **P**.

Output : **Q** the set of concepts that satisfy $\mathbf{C_t}$ and $\mathbf{C_g}$

$\mathbf{H_L} \longleftarrow$ empty()

H and $\mathbf{H_{size}}$ = |**H**| are computed from **r**;

Q $\longleftarrow$ cutting ((**O**, **P**) , **H** , **0** , $\mathbf{H_{size}}$, $\mathbf{H_L}$);

Algorithm 2: Cutting

Input : (**X**,**Y**) a couple of $\mathbf{2^O}$ x $\mathbf{2^P}$, **H** the list of cutters, **I** the number of iterations, $\mathbf{H_{size}}$ the size of **H**, $\mathbf{H_L}$ a set of precedent cutters in left cuttings, $\mathbf{C_t}$ monotonic constraint on **O**, $\mathbf{C_g}$ monotonic constraint on **P**.

Output: **Q** the set of concepts that satisfy $\mathbf{C_t}$ and $\mathbf{C_g}$

(**a,b**) $\longleftarrow$ **H[i]**

If (i≤ $\mathbf{H_{size}}$ **− 1)** // i-th cutter is selected

If ((a ∩ X= ∅) or (b ∩ Y= ∅))

Q $\longleftarrow$ **Q** ∪ cutting **((X,Y),H,i+1,** $\mathbf{H_{size}}$, $\mathbf{H_L}$**)**

Else

If ($\mathbf{C_t}$ **(X\a**) is satisfied)

$\mathbf{H_L} \longleftarrow \mathbf{H_L}$ ∪ (a,b

Q $\longleftarrow$ **Q** ∪ cutting ((**X\a,Y**),**H,i+1,** $\mathbf{H_{size}}$, $\mathbf{H_L}$)

$\mathbf{H_L} \longleftarrow \mathbf{H_L}$ \ (**a , b**)

If ($\mathbf{C_g}$ **(Y\b**) is satisfied ∧ (**a' , b'**) ∈ $\mathbf{H_L}$, **b' ∩ Y\b** ≠ **∅**)

Q $\longleftarrow$ **Q** ∪ cutting ((**X,Y\b**),**H,i+1,** $\mathbf{H_{size}}$, $\mathbf{H_L}$)

Else

Q $\longleftarrow$ (**X,Y**)

Return Q

3.3D FCC Generation by Post-Pruning

In phase 3, 3D frequent patterns are generated by combining each 2D FCP with the heights contributing to its representative slice. However, not all those 3D frequent patterns are FCCs. Some of them are not closed in the height set and should be pruned off. For example, in table 3.6, after combining the first 2D FCP “P1P3 : Q1Q2Q3,2 : 3” with the contributing heights “R2, R3”, a 3D frequent pattern “R2R3 : P1P3 : Q1Q2Q3, 2 : 2 : 3” is generated. This 3D frequent pattern is not a FCC in that it is unclosed in the height set and has a superset “R1R2R3 : P1P3 : Q1Q2Q3, 3 : 2 : 3”(the 4th FCC in the 4th Column of Table 3.6). That is, the 2D FCP is not only contained in slices h2 and h3, but also contained in slice h1. To remove all unclosed 3D frequent closed patterns, we develop a post-pruning strategy based on Lemma 1. If a 2D FCP is contained in other height slices besides its contributing height slices, it is unclosed and hence can be removed; otherwise, it is retained.

Lemma 1*:* Post-pruning Strategy: Let O ' = R ' × P ' ×Q ' be a 3D frequent pattern and R be the whole height set.If ∃R ''∈ (R \ R ') such that ∀Rk ∈ R '', ∀Pi ∈ P ', ∀Qj ∈Q ',Ok,i,j = 1, O ' is unclosed in the height set and can be pruned off; otherwise, O ' is retained.

In the post pruning process, not all relative cells in all non-contributing slices are checked. During the slice checking process, any one cell with value ‘0’ can stop one slice checking. And any slice passing the checking process (all relative cells value ‘1’) without early termination can stop other slice’s checking process in that the pattern is already confirmed to be unclosed.

Table 3.6: RSM Example

Height Set	Representative Slices	2D Frequent Closed Patterns	3D Frequent Closed Cubes
r2,r3	11100 01100 11110 11001	p1p3:q1q2q3,2:3 p1p3p4:q1q2,3:2 p1p2p3:q1q2q3,3:2	r2r3:p1p3p4:q1q2, 2:3:2
r1,r3	11100 11100 11110 00001	p1p2p3:q1q2q3,3:3	r1r3:p1p2p3:q1q2q3, 2:3:3
r1,r2	11101 01100 11110 00101	p1p4:q3q5,2:2 p1p3:q1q2q3,2:3 p1p2p3:q2q3,3:2	r1r2:p1p4:q3q5, 2:2:2
r1,r2,r3	11100 01100 11110 00001	p1p3:q1q2q3,2:3 p1p2p3:q2q3,3:2	r1r2r3:p1p3:q1q2q3, 3:2:3 r1r2r3:p1p2p3:q2q3, 3:3:2

4. PROPOSED SYSTEM

This project aims to address the following limitations of current pattern-based association approaches.

- The first problem is the overwhelmingly large volume of discovered patterns and rules, which is the main obstacle to performance.
- Second, patterns do not carry the structural information of the data themselves; thus, they cannot use this information to improve the interpretation of association rules.
- Finally Representative Slice Mining (RSM) is used to mine FCCs and from that associate rule mining.

5. PROCESS AND IMPLEMENTATION

Mining Association Rules in OLAP Cubes:

On-line analytical processing (OLAP) provides tools to explore data cubes in order to extract interesting information. OLAP is not capable of " explaining relationships that could exist within data. Association rules are one kind of data mining techniques which finds associations among data. A framework is used for mining association rules from data cubes according to a sum-based aggregate measure which is more general than frequencies provided by the COUNT measure".

Data mining techniques such as association rule mining can be used together with OLAP to discover knowledge from data cubes. "The aggregate values needed for discovering association rules are already pre-computed and stored in the data cube. The COUNT cells of a cube store the number of occurrences of the corresponding multidimensional data values. With such summary cells, it is straightforward to calculate the values of the support and the confidence of association rules. The COUNT measure corresponds to the frequency of facts. In an analysis process, users are usually interested in observing multidimensional data and their associations according to measures more relevant than simple frequencies".

A general framework for mining inter-dimensional association rules from multidimensional data. The framework also allows a redefinition of the support and confidence measures based on the SUM aggregate functions over cube indicators (measures). Therefore, the computation of support and confidence according to the COUNT measure becomes a particular case. The general computation of support and confidence of inter-dimensional association rules according to a user defined measure M ∈ M from the mined data cube is defined.

Let a general rule R which complies with the defined inter-dimensional meta-rule :

R: In the context $(\Theta 1, \ldots, \Theta p)$

$(x1 \wedge \cdot \cdot \cdot \wedge xs) \Rightarrow (y1 \wedge \cdot \cdot \cdot \wedge yr)$

The support and the confidence of this rule are therefore computed according to the following general expressions:

$\mathrm{SUPP}(R) = M(x1,\ldots,xs,y1,\ldots,yr,\Theta 1,\ldots,\Theta p,\mathrm{All},\ldots,\mathrm{All})/M(\mathrm{All},\ldots,\mathrm{All},\Theta 1,\ldots,\Theta p,\mathrm{All},\ldots,\mathrm{All})$

$\mathrm{CONF}(R)=M(x1,\ldots,xs,y1,\ldots,yr,\Theta 1,\ldots,\Theta p,\mathrm{All},\ldots,\mathrm{All})/ M(x1,\ldots,xs,\mathrm{All},\ldots,\mathrm{All},\Theta 1,\ldots,\Theta p,\mathrm{All},\ldots,\mathrm{All})$

where $M(x1, \ldots, xs, y1, \ldots, yr,\Theta 1, \ldots,\Theta p, \mathrm{All}, \ldots, \mathrm{All})$ is the sum-based aggregate measure of a sub-cube".

Traditional support and confidence are particular cases of the above expressions which can be obtained by the COUNT aggregation. Nevertheless, in order to simplify notations, support and confidence with the usual terms has to be reffered. Support and confidence are the most known criteria for the evaluation of association rule interestingness. These criteria are the fundamental principles of all Apriori-like algorithms.

However, the large number of rules which are produced may not be interesting.

Let consider again " the association rule $R : X \Rightarrow Y$ which complies with the inter-dimensional meta-rule , where $X = (x1 \wedge \cdot \cdot \cdot \wedge xs)$ and $Y = (y1 \wedge \cdot \cdot \cdot \wedge yr)$ are conjunctions of dimension predicates. Consider a user-defined measure $M \in M$ from data cube C. We denote by PX (respectively PY , PXY) the relative measure M of facts matching X (respectively Y , X and Y) in the subcube defined by the instance $(\Theta1, . . . ,\Theta p)$ in the context dimensions DC. We also denote by $PX = 1 - PX$ (respectively $PY = 1 - PY$) the relative measure M of facts not matching X (respectively Y), i.e., the probability of not having X (respectively Y). The support of R is equal to PXY and its confidence is defined by the ratio PXY PX which is a conditional probability, denoted PY/X, of matching Y given that X is already matched ".

$PX=M(x1,...,xs,All,...,All,\Theta1,...,\Theta p,All,...,All)/M(All,...,All,\Theta1,...,\Theta p,All,...,All)$

$PY=M(All,...,All,y1,...,yr,\Theta1,...,\Theta p,All,...,All)/M(All,...,All,\Theta1,...,\Theta p,All,...,All)$

$PXY=SUPP(R)M(x1,...,xs,y1,...,yr,\Theta1,...,\Theta p,All,...,All)/M(All,...,All,\Theta1,...,\Theta p,All,...,All)$

$PY/X=CONF(R)=M(x1,...,xs,y1,...,yr,\Theta1,...,\Theta p,All,...,All)/M(x1,...,xs,All,...,All,\Theta1,...,\Theta p,All,...,All)$

There are two categories of frequently used evaluation criteria to capture the interestingness of association rules: *descriptive* criteria and *statistical* criteria. In general, one of the most important drawbacks of a statistical criterion is that it depends on the size of the mined population. In addition, it requires a *probabilistic* approach to model the mined population. This approach assumes advanced statistical knowledge of users, which is not particularly true for OLAP users. On the other hand, descriptive criteria are easy to use and express interestingness of association rules in a more natural manner.

Block Diagram:

The block diagram describes how the sample 3D data is extracted and transformed as per the requirement. The taken data is used as input to RSM frame work, the process includes three phases. First phase converts the 3D data into 2D representative slices, D-Miner algorithm is applied on each 2D slice which results a 2D Frequent Closed patterns. Then each 2D FCPs are combines to its contributing heights, thus 3D Frequent Closed Cubes are formed. Finally Association rules are generated on 3D Frequent Closed Cubes.

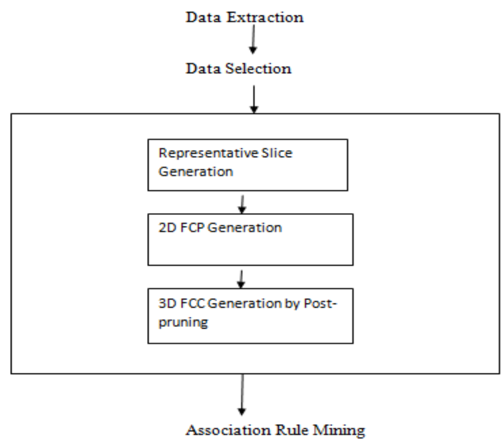
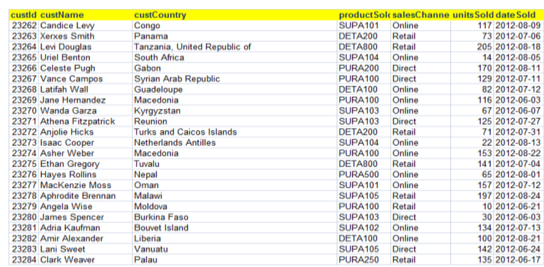


Fig 6.1: Block diagram

6. RESULTS AND DISCUSSIONS

Experimental Results:

Step 1: Input as a sample sales data



custId	custName	custCountry	productSol	salesChanne	unitsSold	dateSold
23262	Candice Levy	Congo	SUPA101	Online	117	2012-08-09
23263	Xerxes Smith	Panama	DETA200	Retail	73	2012-07-06
23264	Levi Douglas	Tanzania, United Republic of	DETA800	Retail	205	2012-08-18
23265	Uriel Benton	South Africa	SUPA104	Online	14	2012-08-05
23266	Celeste Pugh	Gabon	PURA200	Direct	170	2012-08-11
23267	Vance Campos	Syrian Arab Republic	PURA100	Direct	129	2012-07-11
23268	Latifah Wall	Guadeloupe	DETA100	Online	82	2012-07-12
23269	Jane Hernandez	Macedonia	PURA100	Online	116	2012-06-03
23270	Wanda Garza	Kyrgyzstan	SUPA103	Online	67	2012-06-07
23271	Athena Fitzpatrick	Reunion	SUPA103	Direct	125	2012-07-27
23272	Anjolie Hicks	Turks and Caicos Islands	DETA200	Retail	71	2012-07-31
23273	Isaac Cooper	Netherlands Antilles	SUPA104	Online	22	2012-08-13
23274	Asher Weber	Macedonia	PURA100	Online	153	2012-08-22
23275	Ethan Gregory	Tuvalu	DETA800	Retail	141	2012-07-04
23276	Hayes Rollins	Nepal	PURA500	Online	65	2012-08-01
23277	MacKenzie Moss	Oman	SUPA101	Online	157	2012-07-12
23278	Aphrodite Brennan	Malawi	SUPA105	Retail	197	2012-08-24
23279	Angela Wise	Moldova	PURA100	Retail	10	2012-06-21
23280	James Spencer	Burkina Faso	SUPA103	Direct	30	2012-06-03
23281	Adria Kaufman	Bouvet Island	SUPA102	Online	134	2012-07-13
23282	Amir Alexander	Liberia	DETA100	Online	100	2012-08-21
23283	Lani Sweet	Vanuatu	SUPA105	Direct	142	2012-06-24
23284	Clark Weaver	Palau	PURA250	Retail	135	2012-06-17

Fig 7.1: Sample Sales Data

Step 2: RSM FRAMEWORK

Converting data into a Matrix

Step 3: Association Rule Mining

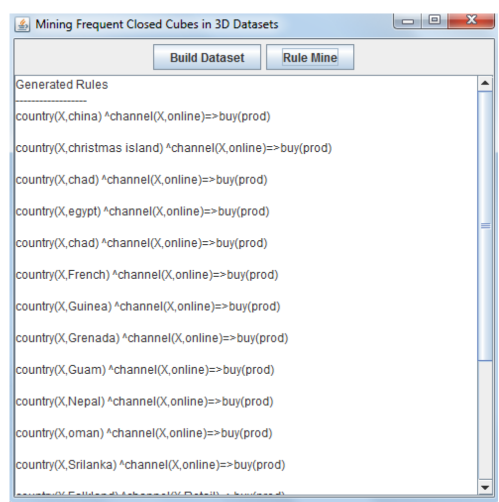


Fig 7.2: Association Rule Mining

Experimental Analysis:

Execution Time for Different data set sizes:

Data Set Size	Execution Time (in milliseconds)
100	150
200	170
300	180
400	188
500	190
600	200
700	205
800	210
900	215
991	225

Table 7.1: Execution time for Different Data Set Sizes

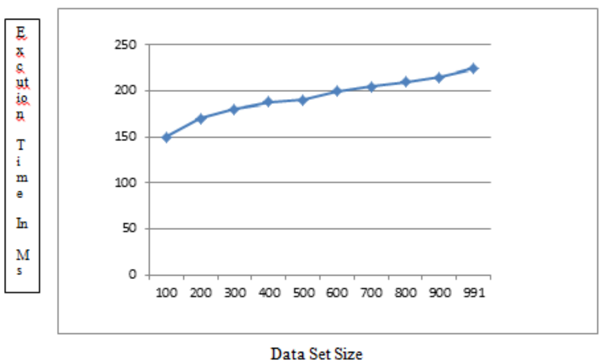


Fig 7.3: Execution time for Different Data Set Sizes

Number of FCCs Generated For Different Data Set Sizes:

Data Set Size	Number of FCCs
100	9
200	18
300	25
400	38
500	46
600	55
700	63
800	72
900	80
991	85

Table 7.2: Number of FCCs generated for Different Data Set Sizes

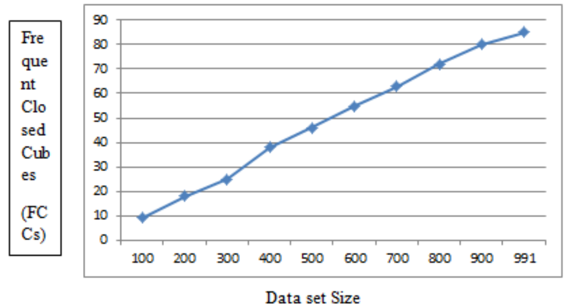


Fig 7.4: Number of FCCs generated for Different Data Set Sizes

7. CONCLUSION

Mining frequent closed cubes for a 3D data is a complex task. So, RSM (Representative Slice Mining) is used to mine Frequent Closed Cubes. In this thesis, 2D frequent closed patterns mined are generalized from 3D context. RSM framework makes us to reuse existing 2D Frequent Closed Pattern mining(FCP) algorithms like D-Miner. D-Miner algorithm is mainly used for dense Boolean datasets. This algorithm generates 2D Frequent Closed Cubes by computing concepts under constraints. Once 2D FCPs are generated, these are combined with their contributing heights. Finally 3D FCCs are generated.

To enhance the flexibility of the existing system, output of 3D FCCs are given as input to the Association Rule Mining. Rules are generated by using minimum support counts. The 3D Association Rule Mining is implemented, which could be made used in classification process.

The system has been implemented several times on different sizes of dataset. Execution time increases when the data set size is increased. RSM framework is efficient when one of the dimensions is considered small.

REFERENCES

[1] Berry, J.-P.Bordat, and A. Sigayret. Concepts cannot afford to stammer. In Proceedings JIM'03, pages 25–35, Metz, France, September 2003.

[2] F. Pan, G. Cong, and A. K. H. Tung. Carpenter: Finding closed patterns in long biological datasets. In SIGKDD'03, pages 637–642, Washington, DC, USA, August 2003.

[3] J. Besson, C. Robardet, and J.-F.Boulicaut.Constraint-based mining of formal concepts in transactional data. In PAKDD'04, pages 615–624,Sydney, Australia, May 2004.

[4] J. Pei, J. Han, and R. Mao. Closet: An efficient algorithm for mining frequent closed itemsets. In Proceedings of 2000 ACMSIGMOD Int. Workshop Data Mining and Knowledge Discovery, May 2000.
[5] Liping Ji, Kian-Lee Tan, Anthony K.H.Tung. Mining frequent closed cubes in 3D datasets. In VLDB '06, September 12-15, 2006,Seoul,Korea.
[6] M. Zaki and C. Hsiao. CHARM: An efficient algorithm for closed association rule mining. In SDM'02, Arlington, VA, USA, April 2002.
[7] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In ACM SIGMOD Conference on Management of Data SIGMOD'93, pages 207–216, 1993.
[8] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and A. I. Verkamo.Fast discovery of association rules. In Advances in Knowledge Discovery and Data Mining, pages 307–328. AAAI Press, 1996.

Authors

R. Komala has received B. Tech, M. Tech in Computer Science and Engineering.
Presently she is working as assistant professor at JNTUK-UCEV, Andhra Pradesh, India.

P.R.S. Naidu has received B. Tech, M. Tech in Computer Science and Engineering.
Presently he is working as assistant professor at JNTUK-UCEV, Andhra Pradesh, India.

N. K. Sumanth has received B. Tech, M. Tech in Computer Science and Engineering.
Presently he is working as assistant professor at JNTUK-UCEV, Andhra Pradesh, India.