Mining Interesting Association Rules with Efficient Data Mining (EDM) Technique

[1] G Neelima, [2] S Vani Kumari
[1]*Assistant Prof. CSE-Dept. GMRIT,Rajam*
[2] *Assistant Prof. CSE-Dept. GMRIT,Rajam*

ABSTRACT

Data mining plays vital role in Business applications. When we consider about the Retail departmental stores, every merchant wants to improve his/her sales and also it becomes a major problem to all merchants because everyone is competeting with other merchants. The solution for this problem is the merchants have to attract the customers towards their shops by placing a up to date inventory in their shops. Once the customer entered into a shop and he has to satisfy with his expected products what he actually wants to buy. Maintaining inventory also a major problem actually it means that placing the products in an order that is once the customer takes a product then definitely the merchant has to get associated products besides that product. So, in order to maintaining the inventory means that giving orders to suppliers what exactly merchant needs and also it would be under an optimized economy or budget. Merchants have to avoid the purchasing the unnecessary products by investing more money. So keeping all these problems in mind we used a technique called MIAR that is Mining Interesting Association Rules. In this merchant can confirm himself that weather the products what he actually expected have association among themselves or not by using support and confidence. .

Keywords

MIAR, ANN.

INTRODUCTION

Data mining plays vital role in Business applications. Consider about the Retail departmental stores, every merchant wants to improve his/her sales and also it becomes a major problem to all merchants because everyone is competeting with other merchants. The solution for this problem is the merchants have to attract the customers towards their shops by placing a up to date inventory in their shops. Once the customer entered into a shop and he has to satisfy with his expected products what he actually wants to buy. Maintaining inventory also a major problem actually it means that placing the products in an order that is once the customer takes a product then definitely the merchant has to get associated products besides that product. So, in order to maintaining the inventory means that giving orders to suppliers what exactly merchant needs and also it would be under an optimized economy or budget. Merchants have to avoid the purchasing the unnecessary products by investing more money. So keeping all these problems in mind we used a technique called MIAR that is Mining Interesting Association Rules. In this merchant can confirm himself that weather the products what he actually expected have association among themselves or not by using support and confidence.

ASSOCIATION ASSOCIATION RULES

Association comes under Descriptive Model and it refers to the data mining task of uncovering relationships among data. An Association Rule is a model that identifies specific types of data associations. These are used to assist retail store management, marketing and inventory control. Association rule mining deals with finding frequent patterns, associations, correlations. An association rule describes the association among items in which when some items are purchased in a transaction, others are purchased too. A transaction *supports* an itemset *Z* , if *Z* is contained in the transaction. The *support for an z itemset* is defined as the ratio of the total number of transactions which support this itemset to the total number of transactions in the database. To make the discussion easier, occasionally, we also let the total number of transactions which support the itemset denote the support for the itemset. The major work of mining association rules is to find all itemsets that satisfy a certain user-specified *minimum support.* Each such itcmsct is rcferred to as *large itemset.*

CONSTRAINT-BASED MINING:

Although there have been many data-mining methodologies and systems developed in recent years, we Contend that by and large, present mining models lack human involvement, particularly in the form of guidance and user control. We believe that data mining is most effective when the computer does what it does best—like searching large databases or counting—and users do what they do best, like specifying the current mining session's focus. This division of labour is best

achieved through constraint-based mining, in which the user provides restraints that guide a search.

1.Mining can also be improved by employing a multidimensional, hierarchical view of the data. Current data warehouse systems have provided a fertile ground for systematic development of this multidimensional mining.

2. Together, constraint-based and multidimensional techniques can provide a more adhoc, query-driven process that effectively exploits the semantics of data than those supported by current stand-alone data-mining systems.

ADHOC AND QUERY DRIVEN:

An ad hoc and query-driven data-mining system can be more effective because it better fits queries to the user's intentions. First, it should offer an ad hoc mining query language, which is a high-level declarative language comparable to the Structured Query Language (SQL) for relational data base management systems. Such a declarative mining language lets users express

• The part of the database to be mined (called the minable view1),

• The type of pattern/rule to be mined, and

• The properties that the patterns should satisfy.

These patterns should include not only numerical constraints on statistical properties (like support, confidence, and correlation), but also those based on attribute domains, classes, and aggregates,1 such as "I.type = 'snacks' and avg(I.price) < 100."Second, a data-mining system should support efficient processing and optimization of mining queries by providing a sophisticated mining-query optimizer.

ESSENTIALS FOR ADHOC DATA MINING:

Constraints are five categories:

•Knowledge type constraints specify the type of knowledge to be mined, such as concept description, association, classification, prediction, clustering, or anomaly. This constraint, unlike other constraints, is usually specified at the beginning of a query because different types of knowledge can require different constraints at later stages.

•Data constraints specify the set of data relevant to the mining task. We often specify such constraints in a form similar to that of an SQL query and process them in query processing.

•Dimension/level constraints confine the dimension(s) or level(s) of data to be examined in a database or a data warehouse. Such constraints follow the model of a multidimensional database and demonstrate the spirit of multidimensional mining. Thus, multidimensional mining can be smoothly incorporated within the framework of constraint-based mining.

•Rule constraints specify concrete constraints on the rules to be mined.

•Interestingness constraints specify what ranges of a measure associated with discovered patterns are useful or interesting from a statistical point of view.

EFFICIENT DATA MINING ALGORITHM:

In this we develop an efficient data mining (EDM) algorithm to generate the interesting association rules according to the user's request. For a user's request, if both the two keywords Antecedent and Consequent are specified with clause and there is no notation"*" specified, then the antecedent and the consequent of the discovered rule will contain only the items specified in < Items >'s after the keywords Antecedent and Consequent, respectively. We call this type of users' requests the Type I request. If the user likes to extract association rules whose antecedent or consequent can contain other items except the items specified in < Items >, then the notation "*" has to be specified in the With clause. We call this type of users' requests the Type II request. The request in which only one of the two keywords Antecedent and Consequent is specified also belongs to the Type I1 request. If both keywords Antecedent and Consequent is the large item generation phase. In this phase, EDM algorithm scans the database to record related information for each interested item and find large items.

The interested items for the Type I request are the items specified in the With clause. The interested items for the Type II and Type III requests are all items in the database. The second phase is the association graph construction phase which constructs an association graph to indicate the associations between every two large items generated in the first phase. The third phase is the interesting large item set generation phase which generates all interesting large item sets by traversing the constructed association graph according to the user's request. The final phase is the interesting association rule generation phase which generates all interesting association rules according to the discovered interesting large item sets, the items specified after the two keywords Antecedent and Consequent, and the user-specified minimum confidence in the user's request.

Large item generation:

In the first phase, algorithm EDM scans the database and builds a bit vector for each interested item. The length of each bit vector is the number of transactions in the database. If an item appears in the ith transaction, the ith bit of the bit vector associated with this item is set to 1. Otherwise, the ith bit of the bit vector is set to 0. The bit vector associated with item z is denoted as BV,. The number of 1's in SV.

Property 1: The support for the itemset { i l , i 2 ,... , i k } is SKI 0 BV,, 0 ... 0 BKk,
where ”d’ is the inner product of two vectors.
Rationale: Because item ij is not a large item, the number of 1’s in the bit vector BV,, is less than the minimum support. Hence, SKI ... BV,, 0 ... e SV,, must be less than the minimum support. The support for the itemset is also less than the minimum support according to the Property 1.So, the itemset is a large itemset. For the Type I request, if there is an interested item which is not a large item, then there is no answer to the request, because any itemset which contains the interested item cannot be a large itemset according to Lemma 1. Otherwise (i.e., all the interested items are large items), the inner products are performed on the bit vectors associated with all the interested items. If the result is no less than the user-specified minimum support, then the set of the interested items is an interesting large itemset.

Consider the example transaction database TDB shown in Table 1. Each record is a <TID, Itemset> pair, where TID is the identifier of the corresponding transaction, and Itemset records the items purchased in the transaction. For example, if a user wants to know if the rule whose antecedent contains only items A and C, and consequent contains only item E is an association rule whose support and confidence achieve 20% and 50%, respectively, then the request is described as follows:
Request 1:
Mining Association Rules
From TDB
With
Antecedent A, C
Consequent E
Support 20%
Confidence 80%

This request belongs to the Type I request and the interested items are A, C and E. Because the minimum support is 20% (i.e., 3 transactions), the interested items A, C and E are all large items, and the associated bit vectors BVA, BVc and BVE are (111111110000000), (11111000111 1001) and (111111100000000), respectively. After performing inner products on BVA, BVc and BVE, the support for the itemset {A, C, E} is 5 (2 3). Hence, the itemset {A, C, E} is an interesting large itemset.

We introduce a data mining language. From the data mining language, users can specify the items in the antecedent and the consequent, and the two criteria: minimum support and minimum confidence of the association rules to be discovered. We propose an efficient data mining algorithm(EDM) to process a user's request.

Association is a rule, which implies certain association relationships among set of objects such as occur together or one implies the other. Goal of Association rule mining helps in finding interesting association relationships among large set of data items. The discovery of such associations can help develop strategies to predict.

An Association Rule is a rule of the format

LHS (left hand side) => RHS (ride hand side).

Let X and Y are two item sets where X, Y $\subseteq$ I (both side contains sets of items) and X$\cap$Y =Ø (don’t share common items).

Briefly, an Association Rule is an expression;

X $\Rightarrow$ Y, where X and Y are set of items.

Each rule is assigned two factors:

- Support
- Confidence

Basic definitions and concepts involved:

Let A= {l_1, l_2, l3 ……l_n} be the set of items. Let T be the set of task relevant data containing database transactions Ti, where Ti is a subset of the items in A.

Support:

A transaction t supports an itemset i if i is contained in t. The support for an itemset i is defined as the ratio of the number of transactions that supports the itemset i to the total number of transactions(or) A transaction t is said to support an item l_i if l_i is present in t. t is said to support a subset of items X$\subseteq$A, if it supports l in X .An item X$\subseteq$A has support s in T denoted by $s(X)_T$, if s% of transactions in T support X.

Support X $\Rightarrow$Y = σ (X$\cup$Y) = s(X$\cup$Y) / no of tuples. (where probability denotes as σ).

Frequent itemset:

If the support for an itemset i satisfies the user-specified minimum support threshold, then i is called frequent itemset, and a frequent itemset of length k a frequent k-itemset.(or) Let T be the transaction database and σ be the user specified minimum support. An itemset X$\subseteq$A is said to be frequent with respect to σ, if $s(X)_T \geq \sigma$.

Confidence:

The confidence of a rule X=>Y is defined as the ratio of the support for the itemsets X U Y to the

support for the itemset X. If itemset Z=XUY is a frequent itemset and the confidence of X=>Y is no less than the user-specified minimum confidence, then the rule X=>Y is an association rule. (OR)The confidence of an association rule is defined as the measure of the probability of an item set dependency on the other item sets in the association rule.
Confidence $X \Rightarrow Y = \sigma (X \mid Y) = s(X \cup Y) / s(X)$.

Example:

The aim of the project is to checking association rule : Let X and Y are two item sets where X, $Y \subseteq I$ (both side contains sets of items) and $X \cap Y = \emptyset$ (don't share common items).Briefly, an Association Rule is an expression;

$X \Rightarrow Y$, where X and Y are set of items.

Support $\geq$ minimum Support

Confidence $\geq$ minimum Confidence

Consider a transaction database consisting of the following transactions:

Transaction id	Items involved
T1	A,B
T2	A,C,D
T3	A,F
T4	A,D,B
T5	C,D,E

Table 1: Association rule generation table

Step1: Find the frequent item sets

{A} =80%, {B}=40%, {C}=40%, {D}=60%,

{A,B}=40%, {A,D}=40%, {C,D}=40%

Step2: Generate Association Rules:

Derive the Association rules with Minimum Support of 40% and Minimum Confidence of 50%

$A \Rightarrow D$ =support $(A \cup D)$/support (A) = 50%

$D \Rightarrow A$ =support $(D \cup A)$/support (D) = 66%

$C \Rightarrow D$ =support $(C \cup D)$/support(C) =100%

$D \Rightarrow C$ = support $(D \cup C)$/support (D) = 66%

$AB \Rightarrow D$ = support $(A \cup B \cup D)$/support (AB) = 50% Similarly other rules, $AD \Rightarrow B$, $AD \Rightarrow C$ can be generated.

CID	Customer sequence
1	<(C)(A,C)(A,C,E)>
2	<(A,E)(A)(A,C,E)(C,E)>
3	<(C)(E)(E)(C,E)>
4	<(B,D)(A,E)(B,C)(A,E)(A,B,E)(F)>
5	<(D)(D,E,F)(C,E,F)(A,D)(B,D)(D,F)>

Table 2 :Customer Sequence Database

DATAMINING LANGUAGE AND DATABASE TRANSFORMATION:

The data mining language is defined as follows. Retailers can query checking of association rules by specifying the related parameters in the data mining language.
Mining <Data Mining Technology>
From <CSD>
With <(D1),(D2), .,(Dm)>
Support <s%>
Confidence <c%>

In the Mining clause, <Data Mining Technology>can be association rules or sequential patterns. In the From clause, <CSD> is used to specify the database name to which users query the association rules or sequential patterns. In the With clause, if the <Data Mining Technology> is <sequential patterns>, <(D1),(D2), .,(Dm)> are userspecified itemsets which ordered by increasing purchasing time, and (Di) can be the notation '*' which represents any sequences. If the <Data Mining Technology> is <association rules>, then m is equal to 2, and D1 and D2 are the itemsets in the antecedent and consequent, respectively, of the discovered rules. Besides, (Di) and the items in Di can be the notation '*' which represents any items.

Support clause is followed by the user-specified minimum support s%.

Confidence clause is followed by the user-specified minimum confidence c% if the <Data Mining Technology> is <association rules>. If the <Data Mining Technology> is <sequential patterns>, this clause is ignored.

In order to find the interesting association rules we need to transform the original transaction data into another type. Each item in each customer sequence is transformed into a bit string. The length of a bit string is the number of the transactions in the customer sequence. If the ith transaction of the customer sequence contains an item, then the i^{th} bit in the bit string for this item is set to 1. Otherwise, the ith bit is set to 0. For example, in Table 1, the bit string for item A in CID 1 is 011. Hence, we can transform the customer sequence database (Table 1) into the bit-string database (Table 2).

From the bit-string database, we can easily compute the number of the transactions in a customer sequence, which contain an itemset. For example, in Table 1, if we want to know how many transactions in CID 1 support the itemset (A,C,E). We can perform logical AND operations on the bit strings for

items A, C and E in CID 1.The number of 1's in the resultant bit string is the number of the transactions which contain the itemset (A,C,E) in CID.
Suppose a customer sequence contains the two sequences S1 and S2. We present an operation called sequential bitstring operation to check if the sequence S1S2 is also contained in this customer sequence. The process of the sequential bit-string operation is described as follows: Let the bit string for sequence S1in customer sequence c is B1, and for sequence S2 is B2. Bit string B1 is scanned from left to right until a bit value 1 is visited. We set this bit and all bits on the left hand side of this bit to 0 and set all bits on the right hand side of this bit to 1, and assign the resultant bit string to a template Tb.
Then, the bit string for sequence S1S2 in c can be obtained by performing logical AND operation on bit strings Tb and B2. If the number of 1's in the bit string for sequence S1S2 is not zero, then S1S2 is contained in customer sequence c. Otherwise, the customer sequence c does not contain S1S2.

For example, consider above Table . We want to check if sequence <(A)(C)> is contained in customer sequence in CID 1. From below Table , we can see that the bit string for items A and C in CID 1 are B_A=011 and Bc=111, respectively, and the template bit string TbZ001. By performing logical AND operation on Tb and Bc, we can obtain that the bit string for sequence <(A)(C)> in customer sequence CID 1 is 001.

Scan the bit-string database once to compute the support for the specified itemset, and then find all the frequent 1-itemsets.For the record in CID i in the bit-string database, if each item in itemset X or X U Y (or Y) is contained in this record,then perform the logical AND operations on the bit strings for the items in itemset X or X U Y (or Y).

The number min of 1's in the resultant bit string is the number of the transactions which contain the itemset X or X U Y (or Y) for CID i. If there is an item in itemset X or X U Y (or Y) not contained in CID i, then m_i is equal to 0. For each item j, if item j is contained in the record in CID i, then find the bit string for item j to count the number Cij of the transactions which contain item j for CID i. Otherwise, the value Cij is 0. Suppose there are p customers and q transactions in the customer sequence database.

CID	Transaction items	Bit string for each item
1	A,C,E	011,111,001
2	A,C,E	1110,0011,1011
3	C,E	1001,0111
4	A,B,C,D,E,F	010110,101010,001000,100000,010110,000001
5	A,B,C,D,E,F	000100,000010,001000,110111,011000,011001

The number of the transactions that contain the itemset X or X U Y (or Y) is m =$\sum_{i=1}^{p}$ mi and the support for the itemset X (or Y) is m/q. If the support is no less than the user-specified minimum support, then compute the support for each item j by the expression $(\sum_{i=1}^{p} Cij)/q$.

Conclusion

In older/ancient days the merchants had invested more money on commodity products and later on invested on other products which were rarely purchased by the customers. Maintaining inventory also a major problem actually it means that placing the products in an order that is once the customer takes a product then definitely the merchant has to get associated products besides that product. So, in order to

maintaining the inventory means that giving orders to suppliers what exactly merchant needs and also it would be under an optimized economy or budget. And also all the merchants had given orders for their products to the suppliers randomly there is no control over the orders. So when the merchants are giving orders to suppliers it should be reasonable. This paper is useful for the merchants. For all these problems MIAR can gives the solution. Suppose some customers purchased some items. When observing the past purchased transactions of all customers and merchant check validation of associated items by using support and confidence of purchased items. After checking validation of association rule merchant can confidently give order for products to the suppliers. So that the investment on products being consumed reasonably. So merchants easily improve their sales by attracting customers towards their shops.

References

[1] Department of Computer Science and *Information* Engineering, Ming Chuan University, Taipei, Taiwan, ROC S.-J. Yen, Y.-S. Lee / Expert Systems with Applications 30 (2006) 650–657

[2] Show-Jane Yen and Arbee L.P. Chen Department of Computer Science National Tsing Hua University Hsinchu, Taiwan 300, R.O.C. *Email:*alpchen@cs.nthu.edu.tw 0-8186-8147-0197 $10.00 *0* 1997 IEEE

[3]http://searchbusinessanalytics.techtarget.com/definition/association-rules-in-data-mining

[4]http://infolab.stanford.edu/~ullman/mining/assocrules.pdf

[5]http://www.nclindia.org/RTI/memorandumbyelaws2005.pdf

[6]http://ieeexplore.ieee.org/xpl/articleDetails.jsp?reload=true&arnumber=617409&contentType=Conference+Publications

[7]http://www.cse.unr.edu/~looney/cs773b/KDDdataMining.pdf

[8]http://wenku.baidu.com/view/3c6a9a3410661ed9ad51f342.html

[9]http://140.134.131.64/wiki/images/d/dd/Association_Rule_and_Quantitative_Mining_among_Infrequent_Items.pdf

[10]http://www.math.upatras.gr/~esdlab/oldEsdlab/en/members/kotsiantis/association%20rules%20kotsiantis.pdf