ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol. 3 , No.1, Pages : 161– 166 (2014)
Special Issue of ICETETS 2014 - Held on 24-25 February, 2014 in Malla Reddy Institute of Engineering and Technology, Secunderabad– 14, AP, India

ANALYZING EDUCATION DATA THROUGH ASSOCIATION RULES: A CASE STUDY

Praveen Kumar Gera [#1], Dr.Sk Altaf Hussain Basha [*2], Katragadda Srinivasa Rao [#3]
[#1] *Asst.Professor , Department of Computer Science and Engineering ,Malla Reddy Inistitute of Engineering and Technology, Hyderabad.India.*
[*2] *Professor , Department of School of computing ,Gokaraju Rangaraju Inistitute of Engineering and Technology, Hyderabad.India.*
[#3] *Asst.Professor , Department of School of computing ,Gokaraju Rangaraju Inistitute of Engineering and Technology, Hyderabad.India.*

[#1] praveenger@gmail.com, [*2] althafbashacse@gmail.com, [#3] vasu.mtech11@gmail.com

Abstract: *The main objective of higher education institutions is to provide quality education to its students. Data mining has attracted a great deal of attention in the information industry in recent years due to the wide availability of huge amount of data and the imminent need for turning such data into useful information and knowledge. An important topic in data mining research is concerned with the discovery of association rules. In this paper, Educational dataset in a university/College is considered for the generation of association rules using Apriori and GRI. The generation association rules are compared and analyzed.*
Keywords: *Data Mining, Association Rules, Frequent Item sets, Apriori, GRI.*

1. INTRODUCTION:

Data mining has attracted popular interest recently, due to the high demand for transforming huge amounts of data found in databases and other information repositories into useful knowledge. The rapid progress in the field owes to the joint efforts of researchers and developers in data mining, data warehousing, database systems, knowledge-base systems, statistics, machine learning, information retrieval, data visualization, high performance computing, and a number of other related fields. Data mining is largely concerned with building models. A model is simply an algorithm or set of rules that connects a collection of inputs (often in the form of fields in a corporate database) to a particular target or outcome. Regression, neural networks, decision trees etc., are some of the techniques for creating models.

There are increasing research interests in using data mining in education. This new emerging field, called Educational Data Mining, concerns with developing methods that discover knowledge from data originating from educational environments . Educational Data Mining uses many techniques such as Decision Trees, Neural Networks, Naïve Bayes, K- Nearest neighbor, and many others. Using these techniques many kinds of knowledge can be discovered such as association rules, classifications and clustering. The discovered knowledge can be used for prediction regarding enrolment of students in a particular course, alienation of traditional classroom teaching model, detection of unfair means used in online examination, detection of abnormal values in the result sheets of the students, prediction about students‟ performance and so on

Many problems of intellectual, economic, and business interest can be phrased in terms of the tasks such as classification, association rules, clustering etc., classification consists of examining the features of a newly presented object and assigning it to one of a predefined set of classes. The objects to be classified are generally represented by records in the database table or a file, and the act of classification consists of adding a new column with a class code of some kind. The classification task is characterized by a well-defined definition of the classes, and a training set consisting of pre classified examples. The task is to build a model of some kind that can be applied to unclassified data in order to classify it. Association rule mining is one of the most important and well researched techniques of data mining. It aims to extract interesting correlations, frequent patterns, associations or casual structures among sets of items in the transaction databases or other data repositories [6].Given a collection of items and a set of records, each of which contain some number of items from the given collection, an association discovery function is an operation against this set of records which return affinities that exist among the collection of items. These affinities can be expressed by rules such as “72% of the records that contain items A,B and C also contain items D and E”. The specific percentage of occurrences (in the case 72) is called the confidence factor of the association. Also, in the association, A,B and C are said to be on an opposite side either side of association. A typical application that can be built using association discovery is supermarket problems. The problem is to analyze customers’

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol. 3 , No.1, Pages : 161– 166 (2014)
Special Issue of ICETETS 2014 - Held on 24-25 February, 2014 in Malla Reddy Institute of Engineering and Technology, Secunderabad– 14, AP, India

buying habits by finding associations between the different items that customers place in their shopping baskets. The gaining insight into matters like "which items are most frequently purchased by customers". It also helps in inventory management, sale promotion strategies, etc. Clustering is the identification of classes (clusters) for a set of unclassified object based on their attributes. The objects are so clustered that the interclass similarities are maximized and the interclass similarities are minimized based on some criteria. Once the clusters are decided, the objects in a cluster are summarized to form the class description [4]. For example, a set of new diseases can be grouped into several categories based on the similarities in their symptoms, and the common symptoms of the diseases in a category can be used to describe that group of diseases.

2. ASSOCIATION RULES:

With widespread applications of computers and automated data collection tools, massive amounts of transaction data have been collected and stored in databases. Discovery of interesting association relationships among huge amounts of data will help in marketing, decision making and business management. Therefore, mining association rules from large datasets has been a focused topic in recent research into knowledge discovery in databases [1, 2, 3].

The following is the standard definition of association rules: Let I= $\{i_1,i_2,\ldots..,i_m\}$ be a set of attribute values, called items. Let D be a set of database transactions where each transaction T is a set of items such that $T \leq I$. An association rule is an implication of the form A=>B, where $A \leq I$, $B \leq I$ and $A \cap B = \varphi$. A is called the antecedent of the rule, and B is the consequent of the rule. The rule A => B holds in the transaction set D with support sup and confidence conf, where sup (A=> B) = P (A U B), conf (A=>B) = P(B/A) = sup(A => B)/sup(A). sup(A) is the percentage of transactions in D that contain A. Rules that satisfy both a minimum support threshold (min_sup) and a minimum confidence threshold (min_conf) are called strong ones. An association rule with cent percent confidence is called an exact association rule.

Association rule mining is a two-step process [5]: (1) The first step consists of finding all frequent itemsets that means each of these itemsets will occur at least as frequently as a predetermined minimum support count. (2) The second step consists of generating strong association rules from the frequent item sets that mean these rules must satisfy minimum support and minimum confidence.

3. ALGORITHMS:

In this paper, Apriori and GRI algorithms are used for generation of association rules.

3.1 Apriori algorithm:

Apriori is an algorithm for extracting association rules from data. It contains the search space for rules by discovering frequent item sets and only examining rules that are made up of frequent item sets. Apriori deals with items and item sets that makes up transactions. Items are a flag-type condition that indicates the presence or absence of a particular thing in a specific transaction. An item set is a group of items which may or may not tend to co-occur within transactions. Apriori proceeds in two stages. Firstly, it identifies frequent item sets in the data, and then it generates rules from the table of frequent item sets.

The first step in Apriori is to identify frequent item sets. A frequent item set is defined as an item set with support greater than or equal to the user-specified minimum support threshold s_{min} . The support of an item set is the number of records in which the item set is found divided by the total number of records. The algorithm begins by scanning the data and identifying the scale-item item sets (I.e. individual items, or item sets of length 1) that satisfy this criterion. Any single item that does not satisfy the criterion is not to be considered further, because adding an infrequent item to an item set will always result in an infrequent item to an item set will always result in an infrequent item set. Apriori then generates item sets recursively using the following steps:

i. Generates a candidate set of length k(containing k items) by combining existing itemsets of length (k-1). For every possible pair of frequent itemsets p and q with length (k-1), it compares the first (f-2) items (in lexicographic order); if they are the same, and the last item in q is (lexicographically) greater than the last item in p, it adds the last item in q to the end of p to create a new candidate item with length k.

ii. Prunes the candidate set by checking every (k-1) length subset of each candidate itemset; all subsets must be frequent itemsets, or the candidate itemset is infrequent and is removed from further consideration.

iii. Calculates the support of each itemset in the candidate set, as support $=N_i/N$ where N_i is the number of records that match the itemset and N is the number of records in the training data.

iv. Itemsets with support $\geq s_{min}$ are added to the list of frequent itemsets.

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol. 3 , No.1, Pages : 161– 166 (2014)
Special Issue of ICETETS 2014 - Held on 24-25 February, 2014 in Malla Reddy Institute of Engineering and Technology, Secunderabad– 14, AP, India

v. If any frequent itemset of length k is found, and k is less than the user-specified maximum rule size k_{max}, it repeats the process to find frequent itemsets of length (k+1).

vi. When all frequent itemsets have been identified, the algorithm extracts rules from the frequent itemsets. For each frequent itemset L with length k>1, Apriori generates rules using the following steps:

vii. Calculates all subsets A of length (k-1) of the itemset such that all the fields in A are input fields and all the other fields in the itemset (those that are not in A) are output fields. Calls the latte subset A^1. (In the first interaction this is just one field, but in later interactions it can be multiple fields).

viii. For each subset A, it calculates the evaluation measure (rule confidence by default) for the rule $A=>A^1$.

ix. If the evaluation measure is greater than the user-specified threshold, it adds the rule to the table, and, if the length k of A is greater than 1, it tests all possible subsets of A with length (k-1).

3.2 Generalized Rule Induction Algorithm:

Generalized rule induction (GRI) generates rules to summarize patterns in the data using a quantitative measure for the interestingness of rules. This measure provides a methods for ranking competing rules and allows the system to contain the search space for useful rules, as well as identifying the best or most interesting rules describing a database.

GRI uses quantitative measure to calculate how interesting a rule may be and uses bounds on the possible values this measure may take a constrain the rule search space. Briefly, the J measure maximizes the simplicity/goodness-of-fit trade-off by utilizing an information theoretic based cross-entropy calculation. A rule in GRI takes the form If Y=y then X=x with probability p.

Where X and Y are two fields (attributes) and x and y are values for those fields. The consequent (the "then" part of the rule) is restricted to being a single values assignment expression while the antecedent (the "if" part of the rule) may be a conjunction of such expressions, for example If Y=y and Z=z, then X=x (with probability p). The complexity of a rule is defined as the number of conjuncts appearing in rule's antecedent.

GRI generates rules through the following steps:

i. It processes each output field Y_i in turn. GRI derives all rules for the current output field before moving on to the next. In other words, GRI uses **depth-first search** to generate to ruleset.

ii. For each output field, it selects each possible output value y_k. Again, processing is depth-first, so all rules predicting the current output field value are generated before the next output field value if considered.

iii. For each output value, it selects each input field X_m.

iv. For each input field, it selects each possible condition x_q. The conditions depend in the type of the input field. For symbolic fields, each value for the field represents a possible condition. For range field, values are stored and each value is tested as a binary split boundary. For each potential split, the J statistic is calculated, and the split with the highest J value is selected as the split for the rule. There are then two possible conditions: greater than the split value, and less than or equal to the split value.

v. For the rule $X_m = x_q => Y_i = y_k$, it computes the j statistic.

vi. If the value of J is greater than the highest J for any rule in the table predicting the same outcome ($Y_i = y_k$) , or if the number of rules in the table is less than the maximum number of rules in the table, and if the maximum support and confidence criteria are met, it inserts the rule in the table (replacing the lower-J rule if necessary) and calculates J_s. Otherwise, it proceeds to the next input field value.

vii. If $J>J_s$, it specializes the rule.

viii. The above steps are to be repeated until all input fields, output field values, and output fields have been considered.

Each rule in the final ruleset has associated instances, support, confidence, and lift values, based on the number of records for which the antecedent and the entire rule are true. Instances is calculated as the number of records for which the antecedent is true. Support is calculated as the instances divided by the total number of records, or $S = N_a/N$ where N_a is the number of records for which the antecedent is true (the instances) a N is the number of records in the training data. Confidence is calculated as the number of records for which the entire rule is true divided by the instances, or $C = N_r/N_a$ where N_r is the number of records for which the entire rule is true. Lift is calculated as the ratio of the conditional probability of the consequent to its unconditional probability.

4. **Data Mining Process**

In present day's educational system, a students" performance is determined by the internal assessment and end semester examination. The internal assessment is carried out by the teacher based upon students" performance in educational activities such as class test, seminar, assignments, general proficiency, attendance and lab work. The end semester examination is one that is scored by the student in semester

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol. 3 , No.1, Pages : 161– 166 (2014)
Special Issue of ICETETS 2014 - Held on 24-25 February, 2014 in Malla Reddy Institute of Engineering and Technology, Secunderabad– 14, AP, India

examination. Each student has to get minimum marks to pass a semester in internal as well as end semester examination.

4.1 Data Preparations

The data set used in this study was obtained from Gokaraju Rangaraju Institute of Engineering and Technology (GRIET) on the sampling method of computer Applications department of course MCA (Master of Computer Applications) from session 2001 to 2011. Initially size of the data is 4500. In this step data stored in different tables was joined in a single table after joining process errors were removed.

4.2 Data selection and transformation

In this step only those fields were selected which were required for data mining. A few derived variables were selected. While some of the information for the variables was extracted from the database. All the predictor and response variables which were derived from the database are given in Table I for reference.

S.No	Attribute	Description	Possible Values
1	PSM	Previous Semester Marks	{First > 60% Second >45 & <60% Third >36 & <45% Fail < 36%}
2	CTG	Class Test Grade	{Poor , Average, Good}
3	SEM	Seminar Performance	{Poor , Average, Good}
4	ASS	Assignment	{Yes, No}
5	GP	General Proficiency	{Yes, No}
6	ATT	Attendance	{Poor , Average, Good}
7	LW	Lab Work	{Yes, No}
8	ESM	End Semester Marks	Third >36 & <45% Fail < 36%}

Table 1: Data Set for Educational Data

The domain values for some of the variables were defined for the present investigation as follows:

a) **PSM –** Previous Semester Marks/Grade obtained in MCA course. It is split into five class values: First – >60%, Second – >45% and <60%, Third – >36% and < 45%, Fail < 40%.

b) **CTG –** Class test grade obtained. Here in each semester two class tests are conducted and average of two class test are used to calculate sessional marks. CTG is split into three classes: Poor – < 40%, Average – > 40% and < 60%, Good –>60%.

c) **SEM –** Seminar Performance obtained. In each semester seminar are organized to check the performance of students. Seminar performance is evaluated into three classes: Poor – Presentation and communication skill is low, Average – Either presentation is fine or Communication skill is fine, Good – Both presentation and Communication skill is fine.

d) **ASS –** Assignment performance. In each semester two assignments are given to students by each teacher. Assignment performance is divided into two classes: Yes – student submitted assignment, No – Student not submitted assignment.

e) **GP -** General Proficiency performance. Like seminar, in each semester general proficiency tests are organized. General Proficiency test is divided into two classes: Yes – student participated in general proficiency, No – Student not participated in general proficiency.

f) **ATT –** Attendance of Student. Minimum 70% attendance is compulsory to participate in End Semester Examination. But even through in special cases low attendance students also participate in End Semester Examination on genuine reason. Attendance is divided into three classes: Poor - <60%, Average - > 60% and <80%, Good - >80%.

g) **LW –** Lab Work. Lab work is divided into two classes: Yes – student completed lab work, No – student not completed lab work.

h) **ESM -** End semester Marks obtained in MCA semester and it is declared as response variable. It is split into five class values: First – >60% , Second – >45% and <60%, Third – >36% and < 45%, Fail < 40%.

4.3 RULE GENARATION:

During the data preparation phase of data mining, it is important to handle the missing values in the our dataset. One preprocessing technique, data cleaning, is applied on the our dataset before generation of association rules. Here all the fields are satisfied with the minimum quality 65%. Hence no field is to be removed from the dataset. The tuples with missing values are removed from the dataset.

a)Association Rules using Apriori Algorithm and GRI Algorithm:

Apriori algorithm and GRI algorithm identifies the frequent item sets (with the given minimum support at 20%) for the our dataset. When all frequent item sets have been identified, the algorithm extracts rules (with the given minimum confidence at 30%) from the generated frequent item sets for the our dataset. Some of the generated association rules by Apriori algorithm are as follows:

Rule 1: for ESM= First Class (1232,100%)

IF PSM = First AND ATT = Good AND CTG = Good THEN ESM = First

Rule 1 specifies that if PSM is First and ATT is Good and CTG is Good then there is 100% chance that it is ESM is First class and 1232 instances support this rule.

Rule 2: for ESM= First Class(1694,100%)

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol. 3 , No.1, Pages : 161– 166 (2014)
Special Issue of ICETETS 2014 - Held on 24-25 February, 2014 in Malla Reddy Institute of Engineering and Technology, Secunderabad– 14, AP, India

IF PSM = First AND CTG = Good AND ATT = Good THEN ESM = First

Rule 2 specifies that if PSM is First CTG=Average and ATT is Good then there is 100% chance that it is ESM is First class and 1694 instances support this rule.

Rule 3 for ESM=Third Class (3489, 59.44%)
IF PSM = Third AND ASS = No AND ATT = Average THEN PSM = Third

Rule 3 specifies that the PSM is third , ASS is NO and ATT is Average, then there is 59.44% chance that it is Third Class and 3489 instances support this rule.

Rule 4 for ESM=First Class (2508, 93.87%)
IF PSM = Second AND ATT = Good AND ASS = Yes THEN ESM = First

Rule 4 specifies that if the PSM is Second , ATT is good and ASS is Yes , then there 93.87% chance that it is First and 2508 instances support this rule.

5. COMPARATIVE STUDY:

Section 4.3 presented the main aspects of two important iterative algorithms used in association rule mining. To be able to compare these algorithms, a suitable comparison framework was established. A Educational dataset, collect from Gokaraju Rangaraju Institute of Engineering and Technology(GRIET), Hyderabad, is used to study the performance of the algorithms.

5.1 Graphical Representation of the Results:

Graph is one of visualization tool with which the user can get an idea very easily. Here, different graphs are generated illustrating the impact of confidence and support on association rules.

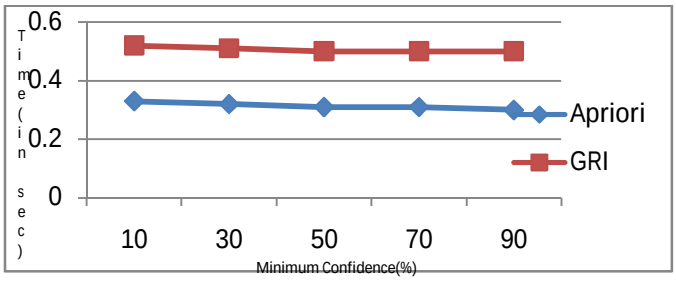


Figure 1: Execution Times of Association Rule derivation Based on Confidence

Figure 1 presents the execution times of algorithms (Apriori, GRI) for different values of the confidence factor (with minimum support at 30%). In figure 1, at 90% minimum confidence, the time taken by Apriori and GRI algorithms to generate association rules is 0.31 seconds and 0.50 seconds respectively. At 10% minimum confidence, the time taken by Apriori and GRI algorithms to generate association rules is 0.38 seconds and 0.53 seconds respectively. From Figure 1, it is observed that GRI has lower performance compared to Apriori.

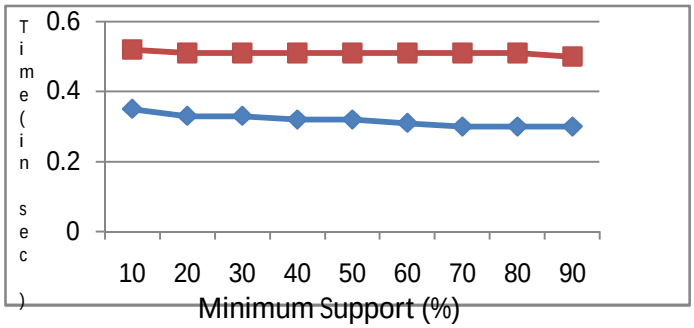
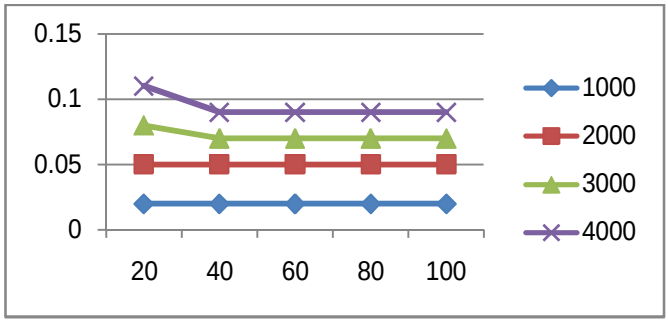


Figure 2: Execution Times of Association Rule Derivation Based on Support

Figure 2 presents the execution times of the algorithms (Apriori, GRI) for different values of the support factor (with the minimum confidence at 30%). In Figure 2, at 10% minimum support, the time taken by Apriori algorithm to generate association rules is 0.39 seconds and at 90% minimum support, the time taken is 0.31 seconds. At 10%minimum support, the time taken by GRI algorithm to generate association rules is 0.55 seconds and at 90% minimum support, the time taken is 0.48 seconds. Figure 2 clearly indicates that GRI has a lower performance compared to Apriori.



Minimum Support (%)

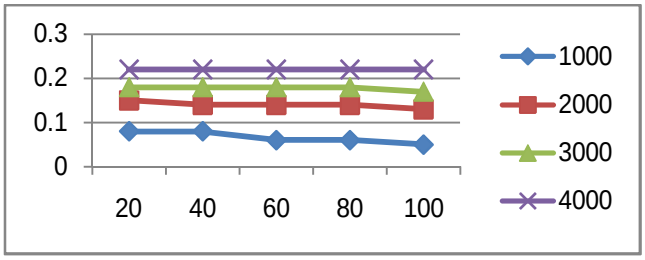
Figure 3: Apriori Scalability by Transactions/Support

Figure 3 presents the execution times of Apriori algorithm for different values of the support factor (with minimum confidence at 30%) on different sized our dataset. In Figure 3, when database size is 1000 records, the time taken by Apriori algorithm to generate association rules for all minimum supports 20%, 40%,60%,80% and 100% is similar i.e., 0.02 seconds. Similarly, when the database size is 2000 records, the time taken by Apriori algorithm to generate

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol. 3 , No.1, Pages : 161– 166 (2014)
Special Issue of ICETETS 2014 - Held on 24-25 February, 2014 in Malla Reddy Institute of Engineering and Technology, Secunderabad– 14, AP, India

association rules for all minimum supports 20%, 40%,60%,80% and 100% is the same i.e., 0.05 seconds. From Figure it is noticed that the performance of the algorithm depends only on the database size but not on support factor.



Minimum Support (%)

Figure 4: GRI Scalability by Transactions/Support

Figure 4 presents the execution time of GRI algorithm for different values of the support factor (with minimum confidence at 30%) on different sized our dataset. Figure 4 illustrates that the performance of the algorithm is depending only on the dimensions of the dataset, the support factor having a very small influence.

5.2 Tabular Form of the Results:

In Table 2 and 3, the minimum confidence and minimum support vary to measure the execution times of the algorithms (Apriori, GRI). When the minimum support is decreasing and minimum confidence is increasing, the number of storing rules and time is also increasing.

Table2: External results with our data by Apriori

Minimum support	Minimum confidence	#strong rules	time in sec s
40%	5199	20%	0.34
30%	6107	30%	0.36
20%	8172	40%	0.36
10%	13844	50%	0.39

Table 3 External results with our data by GRI

Minimum support	Minimum confidence	#strong rules	time in secs
40%	745	20%	0.5
30%	747	30%	0.53
20%	962	40%	0.53
10%	1476	50%	0.55

Table 4 No. of Extract rules Extracted from our dataset

Minimum support	#Exact rules (Apriori)	Exact rules(GRI)
30%	77	636
40%	47	353
50%	33	218
60%	24	93
70%	14	29
80%	13	24
90%	6	9

6. CONCLUSIONS:

Association rules for our dataset are generated using two algorithms Apriori and GRI. The rules are analyzed and some of the important graphs are generated. Using generated rules, the combinations of significant attributes which cause the our data set First Class, Second Class ,Third Class and Fail are for End Semester Marks(ESM) attributes identified. The impact of confidence and support factors on association rules is also discussed.

References

[1] Agrawal R., Imielinski T., & Swami A., Mining association Rules Between Sets of Items in large databases, In Proc. 1993 ACM-SIGMOD International conference on Management of data, pp.207-216, Washington, D.c., (may 1993)

[2] Agrawal R.,& srikanth R., Fact Algorithms for mining Association Rules, In Proc. 1994 Internatonal Conference on very Large data bases, pp.487-499, Santiago, chile, (1994)

[3] Agrawal R.,& srikanth R., mining sequential patterns, In Proc. 1995 International conference on Data Engineering, pp. 3-14, Taipei, Taiwan, March 1995

[4] Fu Y., Discovery of Multiple Level Rules from Large Databases, Ph.D Thesis, SIMON FRASER UNIVERSITY, (1996).

[5] Han J., & Kamber M., Data Mining Concepts & Techniques, Elseiver(2001).

[6] Zhao Q., & Bhowmick S.S., Association Rule mining: A Survey, Technical Report, CAIS, Nanyang Technical university, Singapore, No. 2003116, (2003).