International Journal of Electronics Communication and Computer Engineering
Volume 3, Issue 2, ISSN 2249 –071X

Use of Domain Knowledge In University

Risha Anurag Tiwari
(IPS Academy, Indore)

Abstract **- Data mining is the process of extracting useful information from the huge amount of data stored in the databases. Data mining tools and techniques help to predict business trends those can occur in near future. Association rule mining is an important technique to discover hidden relationships among items in the transaction. In this Work Apriori, Apriori with Domain Knowledge, partitioning and sampling algorithms have been implemented and their performance is evaluated extensively. Apriori algorithm is implemented using K-Way join approach for support counting. Partitioning approach is implemented in the traditional In the case of sampling algorithm the dataset is first partitioned into a number of given partitions and then algorithm is applied by considering one partition as a sample.**

Key Word **– domain, problem description, apriori algorithm, comparative study**

INTRODUCTION

Generally, data mining (sometimes called data or knowledge discovery) is the process of analyzing data from different perspectives and summarizing it into useful information - information that can be used to increase revenue, cuts costs, or both. Data mining software is one of a number of analytical tools for analyzing data. It allows users to analyze data from many different dimensions or angles, categorize it, and summarize the relationships identified. Technically, data mining is the process of finding correlations or patterns among dozens of fields in large relational databases.

Although data mining is a relatively new term, the technology is not. Companies have used powerful computers to sift through volumes of supermarket scanner data and analyze market research reports for years. However, continuous innovations in computer processing power, disk storage, and statistical software are dramatically increasing the accuracy of analysis while driving down the cost.

For example, one Midwest grocery chain used the data mining capacity of Oracle software to analyze local buying patterns. They discovered that when men bought diapers on Thursdays and Saturdays, they also tended to buy beer. Further analysis showed that these shoppers typically did their weekly grocery shopping on Saturdays. On Thursdays, however, they only bought a few items. The retailer concluded that they purchased the beer to have it available for the upcoming weekend. The grocery chain could use this newly discovered information in various ways to increase revenue. For example, they could move the beer display closer to the diaper display. And, they could make sure beer and diapers were sold at full price on Thursdays.

This work comprises study of Apriori, Transaction reduction, Partitioning, Sampling algorithms using Domain knowledge .This work considered elective subject database of near about 7000 records. This database includes information about the student ID, student elective subjects, student immediate goal, college code and percentage marks of the students of a University. When there is provision for students to select electives of their choice during each semester, patterns like combination of electives registered by different students may be useful. Each itemset consist of set of electives. The support of an itemset I is the number of students who have registered the elective subjects present in the set I. If minsup is the minimum support, then I am large if at least minsup number of students have taken all the electives included in I. Association of the form X => Y. Where both X and Y are subsets of electives can be discovered from the course enrollment database in order to discover relationships between elective subjects taken by the students. The system implements an efficient algorithm for discovering such association rules from elective courses enrollment database using domain knowledge.

PROBLEM DESCRIPTION

Association rule mining finds interesting association and/or correlation relationships among large set of data items. Association rules show attributes value conditions that occur frequently together in a given set of data. A typical and widely-used example of association rule mining is Market Basket Analysis

Discovery of association rules are showing attribute-value conditions that occur frequently together in a given set of data. Market Basket Analysis is a modeling technique based on the theory that if you buy a certain group of items then you are more likely to buy another group of items. The set of items a customer buys is referred to as an item set, and market basket analysis seeks to find relationships between purchases. Typically the relationship will be in the form of a rule:

IF {bread} THEN {butter}

This above condition extracts the hidden information i.e. if a customer used to buy bread, he will also buy butter.

The minimum percentage of instances in the database that contain all items listed in a given association rule. There are two types of association rule levels-

1) Support level – The minimum percentage of instances in the database that contain all items listed in a given association rule.

Let T - the set of all transactions under consideration.

S – The support of an item set S is the percentage of those transactions in T which contain S.

|U| and |T| are the number of elements in U and T respectively.

Support (S) = (|U| / |T|) * 100 %

"If A then B" rule confidence is the conditional probability that B is true when A is known to be true.

To evaluate association rules, the confidence of a rule R = "A and B → C" is the support of set of all items that appear in the rule divided by the support of the antecedent of the rule.

Confidence (R) = (support ({A,B,C}) / support ({A,B}))*100%

The confidence of a rule is the number of cases in which the rule is correct relative to the number of cases in which rule is applicable.

An association rule mining algorithm**, Apriori** has been developed for rule mining in large transaction databases. Many variations of the Apriori algorithm have been proposed that focus on improving the efficiency of the original algorithm such as -

- Partitioning (partitioning the data to find candidate itemsets)
- Sampling (Mining on a subset of the given data)

As above we have following tables for elective subject & database around 7000 record:

Table shows the list of subjects in each elective group.

Elective I	MC	ADBMS	CG
Elective II	AJAVA	ERP	AI
Elective III	STIP	CST	SLIP

Table List of Electives

ID	Elective_Sub	Immediate_Goal
A66	MC,AJAVA,STIP	Expanding Knowledge
A67	MC,ERP,STIP	IT Expert
A76	ADBMS,AJAVA,STIP	Software Developer
A77	ADBMS,AJAVA,STIP	Software Developer
A8	ADBMS,ERP,STIP	DBA
C1	MC,ERP,STIP	IT Expert
C10	MC,AJAVA,STIP	Expanding Knowledge
C100	MC,ERP,STIP	IT Expert
C101	MC,ERP,STIP	IT Expert
C102	MC,ERP,STIP	IT Expert
C103	ADBMS,ERP,STIP	DBA
C104	ADBMS,AJAVA,STIP	Software Developer

Table Elective Database

The proposed system has considered following different domain knowledge and found the frequent pattern for elective subjects.

- One student can select only one subject from the list of subjects available in a one elective group, it is obvious that support of candidate belonging to subject of same group is zero.
  Domain Knowledge: - Student can select only one subject from a group.
- Using the result of previous data mining, we can find out rarely used combination. This will be useful for university board of studies for curriculum development
  Domain Knowledge: - The result of previous data mining.
- Some colleges are not offering electives and making single subject as a compulsory subject, remove such records and improve the efficiency of data mining.
  Domain Knowledge: - The college names that are not offering electives.
- If some large itemsets are known in advance, it can be used to generate a set of multiple sized candidate itemsets and reduce the number of database scan.
  Domain Knowledge: - The large itemsets that are known in advance.
- Domain knowledge can be used to restrict uninteresting rules being produced. The basic course of any subject is taken before the advance course of that subject.
  Domain Knowledge: - ADBMS → DBMS, AJAVA → JAVA
- If student is interested to be a "DBA", it suggests which different subjects should select as elective. Domain Knowledge: - Result of previous data mining.

APRIORI ALGORITHM

Apriori is an influential algorithm for mining frequent itemsets for Boolean association rules. The name of the algorithm is based on the fact that the algorithm uses prior knowledge of frequent itemset properties. Apriori employs an iterative approach known as a level-wise search. Where k-itemsets are used to explore (k+1) itemsets.

First the set of frequent itemsets is found. This is denoted as L1.L1 is used to find L2, the set of frequent 2-itemsets, which is used to find L3 and so on, until no more frequent k-itemsets can be found.

The algorithm is as follows

Apriori Algorithm

L1 = { large 1-itemsets }

for (k=2; $L_{k-1} \varnothing \neq$; k++) do begin

C_k = apriori-gen(Lk-1); // New candidates

For all transactions t $\in$ D do begin

C't = subset (C_k, t) // Candidates contained in t

forall candidates c $\in$ Ct do c.count++

end

L_k = {c $\in$ Ct | c.count $\geq$ minsup}

end
Return $\cup k L_k$

Apriori Candidate Generation

apriori-gen(L_{k-1}):
Returns a superset of the set of all large k-items

- First select two itemsets p, q from L_{k-1} s.t. first k-2 items of p and q are the same, form a new candidate k-itemset c as common k-2 items + 2 differing items
- Prune those c, s.t. some (k-1) subset of c is not in Lk-1
- Go thru all transactions in D, increment the counts of all itemsets in Ck
- L_k is the set of all large itemsets in Ck

Apriori algorithm Steps

1. Two pass: first pass counts item occurrences to determine the large 1-itemsets (L_1) from the databases.
2. In subsequent pass, generate a candidate itemset (C_k) by generating a combination of each itemsets and finding a finding a support of each itemsets by scanning databases.
3. Generate a large L_k itemsets by taking a itemsets having a support>= minimum support from the candidate itemsets.
4. Continues steps 2 to 3 until we can't form candidates' itemsets.

Key Points:

- Candidate itemsets are generated using only the large itemsets of the previous pass without considering the transactions in the database.
- The large itemset of the previous pass is joined with itself to generate all itemsets whose size is higher by 1.

Performance Improvement:

- Applying Pruning each generated itemset, that has a subset which is not large, is deleted. The remaining itemsets are the candidate ones. Avoid generating duplicate candidates by ensuring

that the items in each frequent itemset are kept sorted in their lexicographic order.

Bottlenecks of Apriori

- Candidate generation can result in huge candidate sets:
  - 10^4 frequent 1-itemset will generate 10^7 candidate 2-itemsets
  - To discover a frequent pattern of size 100, e.g., {a1, a2, …, a100}, one needs to generate $2^{100} \sim 10^{30}$ candidates.
- Multiple scans of database:
  - Needs (n +1) scans, n is the length of the longest pattern

Apriori With Domain Knowledge

The Apriori algorithm typically identifies the patterns that occur in the whole database. But what if the user is interested in particular attributes for example in our database **Mobile Computing** and wants to check if there is some associational relationship containing the attributes in the database. In such case it is irrelevant to do exhaustive search in the database. The APRIORI WITH DK algorithm includes interaction points for the domain user to give attribute specification if any. The database is then scrutinized according to the specified attribute(s) i.e. the transactions not containing the attributes given by the user are excluded and a working database is created. With this subset of the dataset, the Apriori procedure searches for frequent large itemsets. Although the search dataset is scrutinized but the support for the potential large itemsets is calculated with respect to the original database. The Interactive Association Rule (APRIORI WITH DK) algorithm is presented in Fig

D′ := subset of D containing transactions having the attributes specified by the user.
// (D′ is the working database)
L_1:= {frequent 1-itemsets};
k:=2; //k represents the pass number.
while($L_{k-1} \neq$)
C_k:= new candidates of size k generated from L_{k-1}
for all transactions t□D
increment count of all candidates in C_k that are contained in t
L_k := all candidates in C_k with minimum support k := k+1
Report U_k $L_k$as the discovered frequent itemsets

Methods to improve Apriori Algorithm

Many variation of Apriori algorithm have been proposed that focus on improving the efficiency of the original algorithm that removes the bottlenecks of Apriori algorithm.

Following different methods improves the efficiency of Apriori Algorithm:-

- Hash-based itemset counting**:** A hash-based technique can be used to reduce the size of the candidate k-itemsets, C_k, for k > 1. For example, when scanning each transaction in the database to generate the frequent 1-itemsets, L1, from the candidate 1-itemsets in C1, We can generate all of the 20itemsets for each transaction, hash them into different buckets of a hash table structure, and increase the corresponding bucket counts. A 2-itemset whose corresponding bucket count in the hash table is below the support threshold can not be frequent and thus should be removed from the candidate set.
- Transaction reduction: A transaction that does not contain any frequent k-itemset can not contain any frequent (k+1) – itemsets. Therefore each transaction

can be marked or removed from further consideration since subsequent scans of the database.

- Partitioning: Any itemset that is potentially frequent in Database must be frequent in at least one of the partitions of Database.A partitioning technique can be used that requires just two database scans to mine frequent itemsets.
- Sampling: The basic idea of the sampling approach is to pick a random sample S of the given data D and then search for frequent itemsets in S instead of D. In this way, we trade off some degree of accuracy against efficiency.
- Dynamic itemset counting: A dynamic itemset counting

partitioned into blocks marked by start points. In this

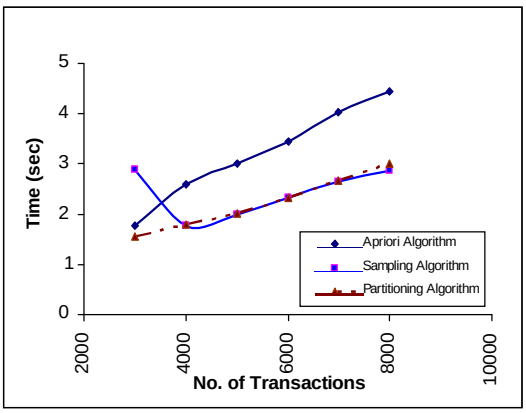
start point, unlike in Apriori, which determines new candidate itemsets only immediately prior to each

that it estimates the support of all the ietmsets that have been counted so far, adding new candidate itemsets if all

COMPARATIVE STUDY

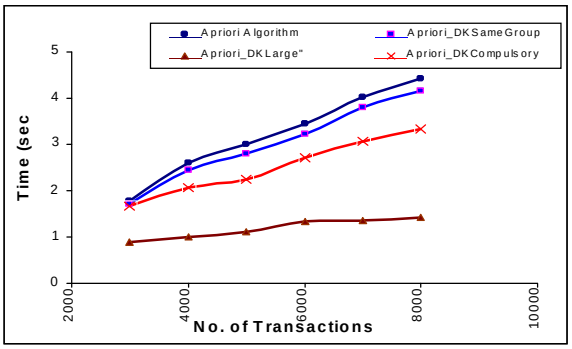
Comparative study of Apriori, Partitioning and Sampling algorithms is carried out for the different sizes of database tabulated in table given below and plotted in the Figure.

No. of Transaction	Apriori	Sampling	Partitioning
3000	1.78	2.9	1.56
4000	2.6	1.78	1.8
5000	3.01	1.99	2.02
6000	3.44	2.32	2.34
7000	4.02	2.64	2.67
8000	4.43	2.87	3



Following table shows time taken by Apriori and Apriori with domain knowledge.

No.Of Transaction	Apriori	Apriori_DK
3000	1.78	1.66
4000	2.6	2.07
5000	3.01	2.25
6000	3.44	2.7
7000	4.02	3.06
8000	4.43	3.34



The study shows that applying domain knowledge on Apriori algorithm takes less time than simple Apriori algorithm and thus improves the efficiency of Apriori algorithm. As the database size will increase the efficiency will also increase.

Apriori - Frequent pattern mining from elective database using Apriori algorithm without any domain knowledge.

Apriori_DKSameGroup – Frequent pattern mining from elective database with domain knowledge that a student can select only one subject from elective group. So the support of subject within a same group will be zero.

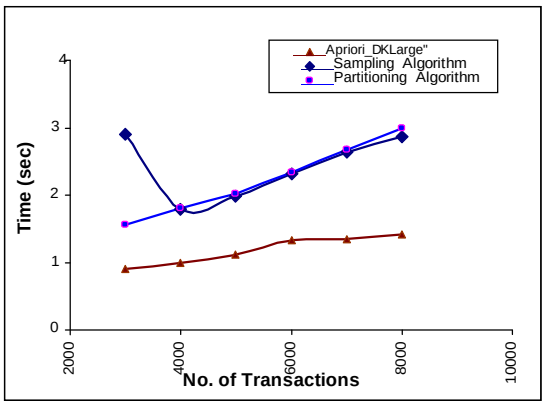


Figure- Comparison of Apriori with Domain Knowledge (Apriori_Large) with Sampling and Partitioning Algorithms

The figure shows that applying domain knowledge to Apriori algorithm is more effective for increasing its efficiency than Sampling and Partitioning algorithm.

Figures shows that Appling domain knowledge on Apriori algorithm always increase the efficiency of data mining but

this is not true always in case of sampling and portioning algorithm. The performance of domain knowledge on these two algorithms is depend on the number transactions and the type of domain knowledge applied. While Apriori improves efficiency of data mining process for all the type of domain knowledge.

Comparison of Apriori, Partition and Sampling algorithm

Sr_No.	Apriori	Partition	Sampling
1	Needs n+1 database scans, n is length of longest pattern	Reduces database scans to one or two	Reduces database scans to two
2	No question of data skew	Negatively impacted by data skew	Negatively impacted by data skew
3	Database has to scan for incremented data.	Incremental generation of rules become easier.	Incremented data is added into original database.
4	Takes 1.38 Seconds to generate results	Takes 1.0 Seconds to generate result.	Takes 0.77 Seconds to generate results

CONCLUSION

The advantage of domain knowledge is that it constrains the search space and rule space thus enhancing the performance of mining process. Use of domain knowledge also reduces the time that experts have to spend on identifying interesting findings and interpreting them. The proposed system implements Simple Apriori Algorithm and also different methods to improve the efficiency of Apriori algorithm using of Domain knowledge.

The drawback of Apriori is that when the cardinality of the longest frequent itemset is k, Apriori needs k passes of database scans. The original Sampling algorithm reduces the number of database scans to one in the best case and two in the worst case. Here the sample is drawn from the database such that it is memory resident. The Partition algorithm reduces the number of database scans to two. By using partitioning, parallel and/or distributed algorithms can be easily created, where each partition could be handled by a separate machine. Incremental generation of association rules may be easier to perform by treating the current state of the database as one partition and treating the new entries as a second partition.The Sampling and Partition algorithm reduce the number of scans of the database to two and have a better performance than Apriori algorithm for large databases.

From the experimental study, it is observed that Sampling and Partitioning reduces pattern generation time than Apriori for the elective subject database.

REFERENCES

[1] George M. Marakas , Modern Data Warehousing, Mining and Visualization, Book Pearson Publication.
[2] David Hand, Heikki Mannila, P Smyth, Principles of Data Mining Book, PHI Publications.
[3] Jiawei Han, Micheline Kamber, Data Mining – concepts and Techniques Book, Elsevier Publications
[4] Data mining and Warehousing S.Prabhu and N.Venkatesan, New age International Publishers.
[5] Data mining group(DMG) homepage http://www.dmg.org
[6] http:/ www.helsinkil\Frequent Itemset and Association Rule Mining Implementation .htm
[7] http://kdd.ics.uci.edu
[8] Rakesh Agrawal and R.Shrikanth,"Fast algorithm for mining association rules", In Proceedings of 20th International conference on very large database.
[9] Computer Sciences Laboratory, RSISE, Australian National University and Advanced Computational Systems CRC, Canberra ACT 0200, Australia
[10] Data Mining Research: Opportunities and Challenges A Report of three NSF Workshops on Mining Large, Massive, and Distributed Data * Robert Grossman**, Simon Kasif, Reagan Moore, David Rocke, and Jeff Ullman
[11] Difficult Training Set : by Helge Grenager Solheim
[12] International workshop on privacy and security issues in data mining
in conjunction with the 8th european conference on principles and practice of knowledge discovery in databases (pkdd04) pisa, italy, september 20, 2004
[13] Catching up with the data: Research issues in data mining streams, Department of Computer Science and Engineering University of Washington
[14] Breaking Out of the Black-Box: Research Challenges in Data Mining Padhraic Smyth Information and Computer Science University of California, Irvine CA 92697{3425 smyth@ics.uci.edu
[15] A view To Discovery From Knowledge-Processing Setsuo Ohsuga Emeritus Professor of University of Tokyo ohsuga@fd.catv.ne.jp
[16] Association Rules Mining On Concept Lattice Using Domain Knowledge, Dept of Computer Science and Technology, Hefei University of Technology, Hefei 230009,China.
[17] A Classification Algorithm Based On Multirelation Domain Knowledge, School of Computer and Information, Hefei University of Technology, Hefei 230009,China.
[18] Savasere E.Omiecinski and S.Navathe. An Efficient algorithm for mining Association Rule in Large Databases. In proceedings of the 21st International conference on very Large Databases, Zurich, Switzerland, September 1995.
[19] Hannu Toivonen. Sampling Large Databases for Association Rules. In Proceedings of the 22nd International conference on Very Large Databases, Bombay, India, September 1996.
[20] Anand S.S.Bell D.A. Huges J.G. The role of Domain Knowledge in Data Mining, In Proceedings of ACM CIKM,USA.

AUTHOR'S PROFILE

Risha Anurag Tiwari
PG. Scholor, IES IPS ACADEMY
Indore, RGPV