Association Rules for Quantitative Data Mining

Viswa Deepak Siingh Baghela[1] , Samar Singh[2] , Archana Gupta[3]

1. Associate Proffesor, at Computer Division IIMT College Of Engineering, Greater Noida
2. M.Tech (CSE) from Teerthankar Mahaveer University
3. M.Tech (CSE) from Teerthankar Mahaveer University

baghela_jinu78@rediffmail.com , chan85samar@gmail.com , archana.archi4u@gmail.com

Abstract: Detatiled elaborations are presented for the idea on two-step frequent itemsets Apriori Algorrithm of Association Rules. over the years, a variety of algorithms for finding frequent item sets in very large transaction databases have been developed. The problems of finding frequent item sets are basic in association rule mining, fast algorithms for solving problems are needed. This paper presents an efficient version of apriori algorithm for mining association rules in large databases to finding maximum frequent itemset at lower level of abstraction. We propose a new, fast and an efficient algorithm with single scan of database for mining complete frequent item sets. To reduce the execution time and increase throughput in new method. Our proposed algorithm works well comparison with general approach of improved association rules. Apriori is the best-known algorithm to mine association rules. It uses a breadth-first search strategy to counting the support of itemsets and uses a candidate generation function which exploits the downward closure property of support. An improved method is called Improved Apriori Algorithm is brought forward owing to the disadvantages of Apriori Algorithm. Moreover,based on Improved Apriori Algorithm,data mining for market-basket analysis is carried out for the relationship between customers' transactions recurrences and products& attributes by making use of SQL Server 2005Analysis Services.
[Viswa Deepak Siingh Baghela, Samar Singh, Archana Gupta. **Association Rules for Quantitative Data Mining.** Academia Arena, 2012;4(1):1-5] (ISSN 1553-992X). http://www.sciencepub.net.

Key words: Data Mining, Apriori Algorithm, Improving Apriori Algorithm, Market Basket .

1. Introduction

In data mining, **association rule learning** is a popular and well researched method for discovering interesting relations between variables in large databases. Piatetsky-Shapir analyzing and presenting strong rules discovered in databases using different measures of interestingness. Data mining, or the efficient discovery of interesting patterns from large collections of data, has been recognized as an important area of database research. The most commonly sought patterns are association rules. Association rule mining is an important data mining technique to generate correlation and association rule. The problem of mining association rules could be decomposed into two sub problems, the mining of large itemsets (i.e. frequent itemsets) and the generation of association rules. Based on the concept of strong rules, Agrawal et al introduced association rules for discovering regularities between products in large scale transaction data recorded by point-of-sale (POS) systems in supermarkets. By using Association rules algorithm to perform market-basket analysis on customers' transactions and also can learn which products are commonly purchased together , and how likely a particular product is is to be purchased along with another. For example, the rule $\{milk,cake\ mix\} \Rightarrow \{frosting\}$ found in the sales data of a supermarket would indicate that if a customer buys milk and cake mix together, he or she is likely to also buy frosting. Such information can be used as the basis for decisions about marketing activities such as, e.g., promotional pricing or product placements. In addition to the above example from market basket analysis association rules are employed today in many application areas including Web usage mining, intrusion detection and bioinformatics.

Market-Basket data mining based on Quantitative Association Rule

In data mining ,association rule learning is a popular and well researched method for discovering interesting relations between variables in large databases. Many algorithms for generating association rules were presented over time. Some well known algorithms are Apriori ,DHP and FP-Growth .Apriori is the best known algorithm to mine strong association rules.

Apriori Algorithm

The problem of association rule mining is defined as: Let $I = \{i_1, i_2, \ldots, i_n\}$ be a set of *n* binary attributes called items. Let $D = \{t_1, t_2, \ldots, t_m\}$ be a set of

transactions called the *database*. Each transaction in *D* has a unique transaction ID and contains a subset of the items in *I*. A *rule* is defined as an implication of the form $X \Rightarrow Y$ where $X, Y \subseteq I$ and $X \cap Y = \emptyset$. The sets of items (for short *itemsets*) *X* and *Y* are called *antecedent* (left-hand-side or LHS) and *consequent* (right-hand-side or RHS) of the rule respectively.

To illustrate the concepts, we use a small example from the supermarket domain. The set of items is *I* = {milk,bread,butter,beer} and a small database containing the items (1 codes presence and 0 absence of an item in a transaction) is shown in the table to the right. An example rule for the supermarket could be $\{\text{butter}, \text{bread}\} \Rightarrow \{\text{milk}\}$ meaning that if butter and bread is bought, customers also buy milk.

Example Of Database with 4 items and 5 transactions.

Transaction Id	Milk	Bread	Butter	Beer
1	1	1	0	0
2	0	0	1	0
3	0	0	0	1
4	1	1	1	0
5	0	1	0	0

To select interesting rules from the set of all possible rules, constraints on various measures of significance and interest can be used. The best-known constraints are minimum thresholds on support and confidence.

- The *support* supp(*X*) of an itemset *X* is defined as the proportion of transactions in the data set which contain the itemset. In the example database, the itemset {milk,bread,butter} has a support of1 / 5 = 0.2 since it occurs in 20% of all transactions (1 out of 5 transactions).
- The *confidence* of a rule is defined $\text{conf}(X \Rightarrow Y) = \text{supp}(X \cup Y)/\text{sup}$. For example, the rule $\{\text{milk}, \text{bread}\} \Rightarrow \{\text{butter}\}$ has a confidence of 0.2 / 0.4 = 0.5 in the database, which means that for 50% of the transactions containing milk and bread the rule is correct.
- Confidence can be interpreted as an estimate of the probability $P(Y \mid X)$, the probability of finding the RHS of the rule in transactions under the condition that these transactions also contain the LHS.
- The lift of a rule is defined as $\text{lift}(X \Rightarrow Y) = \frac{\text{supp}(X \cup Y)}{\text{supp}(Y) \times \text{supp}(X)}$ or the ratio of the observed support to that expected if X and Y were independent. The rule $\{\text{milk}, \text{bread}\} \Rightarrow \{\text{butter}\}$ has a lift of $\frac{0.2}{0.4 \times 0.4} = 1.25$.
- The conviction of a rule is defined as $\text{conv}(X \Rightarrow Y) = \frac{1 - \text{supp}(Y)}{1 - \text{conf}(X \Rightarrow Y)}$. The rule $\{\text{milk}, \text{bread}\} \Rightarrow \{\text{butter}\}$ has a conviction of $\frac{1 - 0.4}{1 - .5} = 1.2$, and can be interpreted as the ratio of the expected frequency that X occurs without Y (that is to say, the frequency that the rule makes an incorrect prediction) if X and Y were independent divided by the observed frequency of incorrect predictions. In this example, the conviction value of 1.2 shows that the rule $\{\text{milk}, \text{bread}\} \Rightarrow \{\text{butter}\}$ would be incorrect 20% more often (1.2 times as often) if the association between X and Y was purely random chance.
- The property of succinctness(Characterized by clear, precise expression in few words) of a constraint. A constraint is succinct if we are able to explicitly write down all Item-sets,that satisfy the constraint.

Example : Constraint C = S.Type = {NonFood}
Products that would satisfy this constraint are for ex. {Headphones,Shoes,Toilet paper}

Process

Association rules are usually required to satisfy a user-specified minimum support and a user-specified minimum confidence at the same time. Association rule generation is usually split up into two separate steps:

1. First, minimum support is applied to find all *frequent itemsets* in a database.
2. Second, these frequent itemsets and the minimum confidence constraint are used to form rules.

Notation and Basic Concepts

The most common frame-work in the association rule generation is the "Support-Confidence" one. In [13], authors considered another frame-work called correlation analysis that adds to the support-confidence. In this paper, they combined the two phases (mining

frequent itemsets and generating strong association rules) and generated the relevant rules while analyzing the correlations within each candidate itemset. This avoids evaluating item combinations redundantly. Indeed, for each generated candidate itemset, they computed all possible combinations of items to analyze their correlations. At the end, they keep only those rules generated from item combinations with strong correlation. If the correlation is positive, a positive rule is discovered. If the correlation is negative, two negative rules are discovered.

Let _ = {*i*1, *i*2 … *im*} be a universe of items. Also, let *T* = {*t*1, *t*2 …tn} be a set of all transactions collected over a given period of time. To simplify a problem, we will assume that every item *i* can be purchased only once in any given transaction *t*. Thus *t* _ _ ("t is a subset of omega"). In reality, each transaction *t* is assigned a number, for example a transaction id (TID).

Support

The *support* of an itemset is the fraction of the rows of the database that contain all of the items in the itemset. Support indicates the frequencies of the occurring patterns. Sometimes it is called *frequency*. Support is simply a probability that a randomly chosen transaction *t* contains both itemsets *A* and *B*.

Confidence

Confidence denotes the strength of implication in the rule. Sometimes it is called *accuracy*. Confidence is simply a probability that an itemset *B* is purchased in a randomly chosen transaction *t* given that the itemset *A* is purchased In general, a set of items (such as the antecedent or the consequent of a rule) is called an itemset. The number of items in an itemset is called the length of an itemset. Itemsets of some length k are referred to as k-itemsets. Generally, an association rules mining algorithm contains the following steps:

Quantitative rule mining approaches

Adaptation of the APRIORI algorithm for mining quantitative association rules was identified shortly after the introduction of APRIORI algorithm, the necessity for quantity in mining association rules was first identified in [14]. It proposed rules of the form x=qx=>y=qy i.e. it associated a single quantity q to the antecedent and the consequent. This was done by decomposition of one quantitative attribute into several binary attributes. In almost all works dealing with mining quantitative attributes, discretization is considered as the tool for reducing the time complexity associated with mining quantitative association rule mining algorithms as the number of quantities can be infinite. Discretization was first proposed in [16]. Mere reduction of quantitative values into Boolean values was also proposed by some authors [11][15]. In [7] it was argued that discretization leads to information loss and hence completely omitting discretization step in mining QAR was proposed. It proposed a representation of the rules based on half-spaces. But the rules generated with such method are different from the classical rules and their understandability is questioned. A new measure of quality for mining association rules is proposed in [9]. Here a new kind of rule called ordinal association rule is used to mine QAR, it removes the step of discretization and complete disjunctive coding and aims at obtaining variable discretization of numerical attributes. Usage of statistical values, like mean as the measure of quality for mining quantitative association rules was proposed in [8] and [12]. The time complexity of QAR mining increases exponentially as the number of possible attributes values grows. This time consumption is another important and discussed issue addressed mainly in [12] and [13]. Quantitative attributes result in lots of redundant rules, most algorithms generate rules that provide almost the same identical information. Such redundancy issue has been partially mentioned in [10], where optimized support and confidence measures are defined and used.

Improved Apriori Algorithm

In Apriori algorithm all the candidate itemsets with the same length must be stored in the memory ,which results in a waste of space. To generate large itemsets, the database passed as many times as the length of the longest large itemsets. Namely, the database is scanned and the support of each candidate itemsets is counted after the new candidate itemsets are generated , which results in a waste of time for large database. This is the performance bottleneck of Apriori Algorithm.

The basic idea of Improved Apriori Algorithm is proposed according to the above deficiencies. In the Improved algorithm , which is fundamentally different from Apriori , we need not store all candidate itemsets in the memory and pass over the database only once. Find out all the high frequency 1-diamensional data itemsets L_1 and then L_1 is used to identify all the high frequency 2-diamensional data itemsets L_2, what's more , use L_2 to find C_2 , the rest may be deduced by analogy until no new high frequency itemset exist. Te realization from L_{k-1} to L_k is connecting L_{k-1} and its own to generate a candidate set of k-dimensional set of data itemsets, denoted by C_k ,and then counting the frequency of C_k's data itemsets, discarding low-frequency data itemsets, forming L_k . The connecting process is taking out p and q from L_{k-1}. If p and q are the same as the pre-k-2 items, make a connection (S. Muggleton 1992). The Improved function apriori-gen is as follows.

Procedure
apriori_gen (L_{k-1}:frequent (k-1)_item sets; minsup)
for each itemset pϵ L_{k-1}
for each itemset qϵ L_{k-1}
if ($p.item_1$=$q.item_1$) ∧ ($p.item_2$=$q.item_2$) ∧…..∧ ($p.item_{k-2}$=$q.item_{k-2}$) then
{ c= p∪q
for each itemset pϵ L_{k-1} //scan all elements of L_{k-1}
for each itemset cϵ C_k //scan all elements of C_k
if p is the subset of c then
c.count++;
C_k= {cϵ C_k | c.count =k};
}
Return C_k;

In order to reduce the size of candidate sets , the improvement is set proposed. The improved algorithm has the excellent property that the database is not used repeatedly. Obviously the improved algorithm is superior when the number of data itemsets continuously increases.

Group items into higher conceptual groups, e.g. white and brown bread become "bread."Reduce the number of scans of the entire database (Apriori needs n+1 scans, where n is the length of the longest pattern)

- Partition-based apriori
- Take a subset from the database, generate candidates for frequent itemsets; then confirm the hypothesis on the entire database.

Analysis of the mining results

Realize Association Rules algorithm by making use SQL Server 2008 Analysis Services. Association Rules are brought forward.

1- Probability is put to use instead of Confidence.
2- How to calculate the importance of Association Rules?

$$IMPORTANCE_{A->B}=\log\frac{p(B|A)}{p(B|notA)}$$

3- Set the parameters of the algorithm.

The mining rules are shown above ,which sort on the basis of importance and probability of association.

CONCLUSION

An Improved Apriori Algorithm is proposed to reduce the size of candidate sets by studying on Apriori Algorithm of Association Rules and the deficiencies of Apriori Algorithm. Conclusions are made on association rules between product recurrence and other attributes by doing data mining using SQL Server 2008 Analysis Services.

FUTURE SCOPE

The work presented in this paper points to several directions for future research. A natural next step is to experiment with other kinds of mining operations (e.g. clustering and classi_cation [8]) to verify if our conclusions about associations hold for these other cases too. We experimented with generalized association rules [22] and sequential patterns [23] problems and found similar results. In some ways associations is the easiest to integrate as the frequent itemsets can be viewed as generalized group-bys. Another useful direction is to explore what kind of a support is needed for answering short, interactive, adhoc queries involving a mix of mining and relational operations. Howmuch can we leverage from existing relational engines? What data model and language extensions are needed? Some of these questions are orthogonal to whether the bulky mining operations are implemented using SQL or not. Nevertheless, these are important in providing analysts with a well-integrated platform where mining and relational operations can be inter-mixed in flexible ways.

Correspondence to:
Viswa Deepak Siingh Baghela[1] , Samar Singh[2] , Archana Gupta[3]
Associate Proffesor, at Computer Division IIMT College Of Engineering, Greater Noida
M.Tech (CSE) from Teerthankar Mahaveer University
M.Tech (CSE) from Teerthankar Mahaveer University
Cellular phone: 091-09911722708;
091-08010407012
Emails: baghela_jinu78@rediffmail.com ; , chan85samar@gmail.com;archana.archi4u@gmail.com

REFERENCES

[1] R. Agrawal, A. Arning, T. Bollinger, M. Mehta, J. Shafer, and R. Srikant. The Quest Data Mining System. In Proc. of the 2nd Int'l Conference on Knowledge Discovery in Databases and Data Mining, Portland, Oregon, August 1996.
[2] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases.
In Proc. of the ACM SIGMOD Conference on Management of Data, pages 207{216, Washington, D.C., May 1993.
[3] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and A. I. Verkamo. Fast Discovery of Association Rules. In U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, Advances in Knowledge Discovery and Data Mining, chapter 12, pages 307{328.AAAI/MIT Press, 1996.
[4] R. Agrawal and J. Shafer. Parallel mining of association rules. IEEE Transactions on Knowledge and Data Engineering, 8(6), December 1996.

[5] R. Agrawal and K. Shim. Developing tightly-coupled data mining applications on a relational database system. In Proc. of the 2nd Int'l Conference on Knowledge Discovery in Databases and Data Mining, Portland, Oregon, August 1996.
[6] S. Brin, R. Motwani, J. D. Ullman, and S. Tsur. Dynamic itemset counting and implication rules for market basket data. In Proc. of the ACM SIGMOD Conference on Management of Data, May 1997.
[7] D. Chamberlin. Using the New DB2: IBM's Object Relational Database System. Morgan Kaufmann, 1996.
[8] Y. Aumann and Y. Lindell. A statistical theory for quantitative association rules. In: Journal of Intelligent Information Systems, 20:255_283, 2003.
[9] S. Guillaume. Discovery of ordinal association rules. In: Proceedings of the Sixth Pacific-Asia Conference PAKDD'02, Taiwan, 2002.
[10] R. Rastogi and K. Shim. Mining optimized association rules with categorical and numeric attributes.Proc. IEEE Trans. on KD Engineering, 14(1), 2002.
[11] S. Imberman and B. Domanski. Finding association rules from quantitative data using data booleanization. In: Proceedings of the Seventh Americas Conference on Information Systems (AMCIS 2001), 2001.
[12] G.I. Webb. Discovering associations with numeric variables .In: Proc. of ACM SIGMOD Conference on Management of Data, San Francisco, CA, 2001.
[13] J. Wijsen and R. Meersman. "On the complexity of mining quantitative association rules.". In: Data Mining and Knowledge Discovery, 2:263_281, 1998.
[14] R.J. Miller and Y. Yang." Association rules over interval data.". In: Proc. of ACM SIGMOD Conference on Management of Data, Tuscon, AZ, 1997.
[15] T. Fukuda, Y. Morimoto, S. Morishita, and T. Tokuyama. Mining optimized association rules for numeric attributes. In: Proc. of ACM SIGMOD Conference on Management of Data, Montreal, Canada, 1996.
[16] Srikant, R., and. Agrawal, R. " Mining quantitative association rules in large relational databases". In: Proc. of ACM SIGMOD Montreal, 1996.
[17] U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors. Advances in Knowledge Discovery and Data Mining. AAAI/MIT Press, 1996.
[18] J. Han, Y. Fu, K. Koperski, W. Wang, and O. Zaiane. DMQL: A data mining query language for relational datbases. In Proc. of the 1996 SIGMOD workshop on research issues on data mining and knowledge discovery, Montreal, Canada, May 1996.
[19] K. Rajamani, B. Iyer, and A. Chaddha. Using DB/2's object relational extensions for mining associations rules. Technical Report TR 03,690., Santa Teresa Laboratory, IBM Corporation, sept 1997.
[20] S. Sarawagi, S. Thomas, and R. Agrawal. Integrating association rule mining with relational database systems: Alternatives and implications. Research Report RJ 10107 (91923), IBM Almaden Research Center, San Jose, CA 95120, March 1998. Available from http://www.almaden.ibm.com /cs/quest.
[21] R. Srikant and R. Agrawal. Mining Generalized Association Rules. In Proc. of the 21st Int'l Conference on Very Large Databases, Zurich, Switzerland, September 1995.

12/12/2011