MIC Framework: An Information-Theoretic Approach to Quantitative Association Rule Mining

Yiping Ke James Cheng Wilfred Ng
Department of Computer Science
Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong, China
{keyiping, csjames, wilfred}@cs.ust.hk

Abstract

We propose a framework, called MIC, which adopts an information-theoretic approach to address the problem of quantitative association rule mining. In our MIC framework, we first discretize the quantitative attributes. Then, we compute the normalized mutual information between the attributes to construct a graph that indicates the strong informative-relationship between the attributes. We utilize the cliques in the graph to prune the unpromising attribute sets and hence the joined intervals between these attributes. Our experimental results show that the MIC framework significantly improves the mining speed. Importantly, we are able to obtain most of the high-confidence rules and the missing rules are shown to be less interesting.

1. Introduction

Quantitative association rules (*QARs*) have served as a useful tool to discover association relationships among a set of attributes in business and scientific domains. In a QAR, attributes are not limited to being boolean but can be quantitative (e.g. age, salary) or categorical (e.g. sex, brand). Each quantitative attribute is associated with an interval of numeric values, while each categorical attribute is associated with a category. Thus, QARs are more expressive and informative than *Boolean Association Rules* (*BARs*) [2], which only consider the presence of the attributes. An example of a QAR in an employee database is: $\{$`age[25, 40]`, `sex[female]`$\} \Rightarrow \{$`salary[2000, 2500]`$\}$ (`sup` = 3%, `conf` = 80%). This QAR states that "3% (*support*) of the employees are females aged between 25 and 40 and earning a salary of between $\$2,000$ and $\$2,500$", while "$80\%$ (*confidence*) of females aged between 25 and 40 are earning a salary of between $\$2,000$ and $\$2,500$".

Given a database, a *minimum support threshold* and a *minimum confidence threshold*, the problem of *QAR mining* [7] is to find all QARs with support and confidence no less than the given thresholds.

A general approach to the QAR mining problem is to map it into the problem of conventional BAR mining [2, 3]. The idea is that, for each distinct *value* of a quantitative or categorical attribute, the pair $\langle$*attribute*, *value*$\rangle$ is mapped to a boolean attribute. However, in many cases, the domain of a quantitative attribute can be very large. In [7], Srinkant and Agrawal propose to discretize the domain of a quantitative attribute into intervals and combine consecutive base intervals to gain sufficient support. Then, each $\langle$*attribute*, *interval*$\rangle$ pair of the quantitative attribute is mapped to a boolean attribute. Finally, an Apriori-like algorithm is employed to compute the frequent itemsets. Hereafter, we denote their mining approach as *SA* and use it as a baseline to evaluate the performance of our approach. We note that other existing studies [10, 8] have also discussed the discretization techniques.

However, QAR mining on a discretized database is still expensive due to the following two combinatorial explosion problems of QAR mining. The same as BAR mining, QAR mining also suffers from the problem of combinatorial explosion of attribute sets; that is, given $\mathcal{N}$ unique attributes, the number of non-empty attribute sets is $2^{\mathcal{N}} - 1$. Although in practice, the number of distinct attributes in a QAR mining problem may not be large, combining the consecutive intervals of a quantitative attribute leads to another combinatorial explosion problem: if the domain of a quantitative attribute is partitioned into n intervals, the total number of intervals of that attribute after combining the consecutive intervals is $\mathcal{O}(n^2)$. When we further join the attributes in the mining process, the number of itemsets (i.e. a set of $\langle$*attribute*, *interval*$\rangle$ pairs) can become prohibitively large due to the large number of intervals associated with an attribute. For example, it is common for a quantitative attribute to have more than 200 intervals in a QAR mining problem; however, there are $(200 * (200+1)/2)^2 = 404,010,000$ different combinations of intervals for only two such attributes, which is equivalent to 404,010,000 candidate attribute sets in a BAR mining problem. This number further grows when more than two attributes are joined.

In this paper, we adopt an information-theoretic approach and propose a framework, called MIC, to mine QARs. MIC prunes the original prohibitively large search space of QAR mining by removing the parts of the search space that reflect the insignificant informative-relationships between the attributes, thereby greatly improving the mining efficiency. Our extensive experiments show that compared with SA, MIC improves the mining speed by up to two orders of magnitude. Moreover, MIC is able to obtain most of the rules that have high confidence and we justify that the missing rules are of less interestingness.

2. The MIC Framework

The MIC framework seamlessly incorporates the *mutual information* concept from information theory [5] into the context of QAR mining. There are three main phases in the MIC framework, as shown in Figure 1.

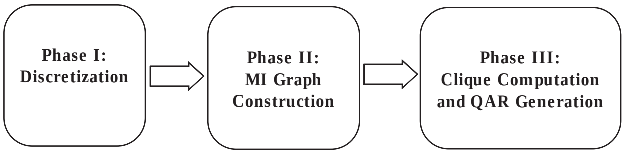


Figure 1. Three Phases of the MIC Framework

Phase I: Discretization. The domain of each quantitative attribute is partitioned into a set of *base intervals* by applying an *unsupervised* discretization technique. The base intervals are then labelled with a set of consecutive integers such that the order of the base intervals is preserved. During the mining process, each base interval is considered as an indivisible unit while consecutive base intervals may be combined into larger intervals. We also label the values of a categorical attribute with a set of consecutive integers.

Phase II: MI Graph Construction. Based on the discretized database derived from Phase I, MIC computes the *normalized mutual information* between each pair of attributes.

The normalized mutual information between two attributes x and y, denoted as $\widetilde{I}(x;y)$, is defined by

$$\widetilde{I}(x;y) = \frac{I(x;y)}{I(x;x)},$$

where $I(x;y)$ is the mutual information between x and y. The semantics of the normalized mutual information is the percentage of reduction in uncertainty about x due to the knowledge of y. Thus, the value of the normalized mutual information falls within the unit interval $[0,1]$. To compute the normalized mutual information between each pair of attributes, we only need to scan the database once.

Given a predefined *minimal mutual information threshold* μ, we then construct a *mutual information graph (MI graph)*, which is a directed graph, $G_{MI} = (V, E)$, where the set of vertices V is the set of all attributes and the set of directed edges $E = \{(x_i, x_j) : \widetilde{I}(x_i; x_j) \geq \mu\}$. We say that a pair of attributes, x_i and x_j, has a *strong informative-relationship* if $\widetilde{I}(x_i; x_j) \geq \mu$.

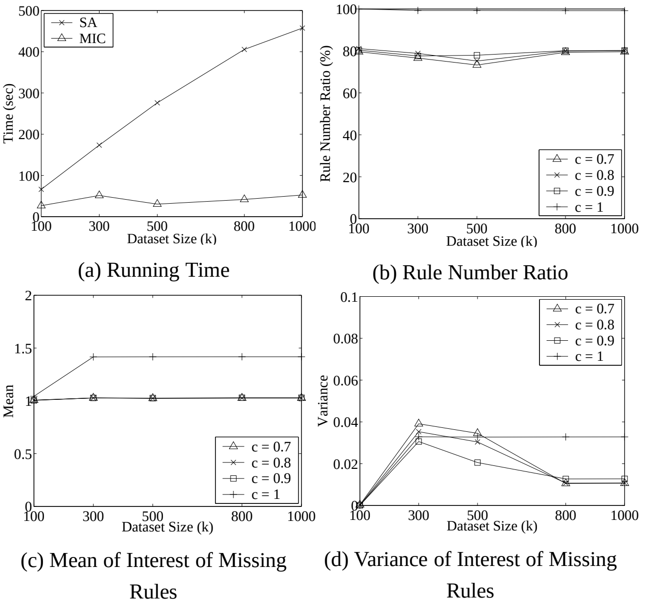
Phase III: Clique Computation and QAR Generation. We find all the cliques in the MI graph G_{MI} and simultaneously compute the set of frequent itemsets based on the cliques using a prefix tree structure. We use the technique of *diffset* [9] on the prefix tree to compute the frequency of the itemsets, so that we only scan the database twice: one for computing the normalized mutual information and the frequent items, and the other for computing the initial diffsets (i.e. sets of transaction IDs). All other frequent itemsets are then computed using the diffsets. We then generate the QARs from the frequent itemsets using the rule generation technique for BARs [3].

Note that we capture the relationship between the attributes which form a QAR using the normalized mutual information. Since the MI graph is built by discarding the insignificant informative-relationships between the attributes, a clique in the MI graph corresponds to the set of attributes, which potentially form a QAR and hence a frequent itemset. Therefore, by finding all the cliques in the MI graph, we are able to obtain all or most of the frequent itemsets and then the QARs.

We can view the QAR mining problem at two conceptual levels: the *attribute level* that consists of the attributes and the *interval level* that consists of the corresponding intervals of the attributes. SA directly operates on the interval level throughout the whole mining process because its pruning is performed on the intervals of the attributes. In contrast, MIC performs pruning first at the attribute level. All pairs of attributes that do not have a strong informative-relationship are not chosen to form an itemset and consequently all their intervals are also pruned. Meanwhile, MIC also performs pruning at the interval level by the apriori property as does SA. Thus, the search space of MIC is significantly smaller than that of SA. Although the pruning at the attribute level may lose some QARs in the final result, we emphasize that our method is not an approximation technique that improves the efficiency at the expense of accuracy. Instead, we show that MIC not only significantly outperforms SA, but the set of rules it obtained is also of higher quality than that obtained by SA, since the attributes appearing in the same rule are informatively dependent on each other. More importantly, the missing QARs of MIC are less interesting as shown by our experiments.

3. Experimental Results

We evaluate the performance of our MIC framework on both synthetic and real datasets, using SA as the baseline for comparison. The synthetic datasets are generated by the IBM Quest Synthetic Data Generation Code [1], and the real datasets are chosen from the commonly used UCI machine learning repository [6]. In order to make a fair comparison, we apply *equidepth*, which is also used in SA, to discretize the database. Equidepth partitions the domain of a quantitative attribute into n base intervals so that the frequency (i.e. the number of transactions) of each base interval is roughly the same. We also employ another well-established measure, *interest* [4], for the interestingness of an association rule to assess the quality of the QARs obtained and missed. Due to the limited space, we only present the results for synthetic datasets.



(a) Running Time

(b) Rule Number Ratio

(c) Mean of Interest of Missing Rules

(d) Variance of Interest of Missing Rules

Figure 2. Running Time, Rule Number Ratio, Mean and Variance of Synthetic Datasets

Figure 2(a) reports the running time for generating frequent itemsets. MIC runs significantly faster than SA and the improvement in speed increases linearly as the size of the dataset increases. Figure 2(b) shows the ratio of the number of rules obtained by MIC to that obtained by SA. On average, MIC obtains 80% of rules that have a confidence over 0.7, while it obtains almost all rules that have a confidence of 1. However, we justify in Figures 2(c-d) that the mean of the interest of the missing rules is almost 1 in all cases, with an extremely small variance of at most 0.04. According to Brin et al. [4], if the *interest* of a rule $X \Rightarrow Y$ is 1, then X and Y are independent. Therefore, we can conclude that the rules missed in MIC are of little significance by the *interest* measure.

4. Conclusions

In this paper, we present the MIC framework that adopts an information-theoretic approach to mine the QARs. We propose a normalization to the mutual information concept and then apply it to model the informative-relationships between the attributes in a QAR mining problem. Based on the normalized mutual information, we construct an MI graph that captures the strong informative-relationships between the attributes. We find that the cliques in the MI graph correspond to the potential frequent itemsets in the mining problem. We incorporate the enumeration of the cliques seamlessly into the mining process to compute the frequent itemsets. The clique enumeration limits the mining process to a much smaller but more relevant search space, thereby significantly improving the mining efficiency. Our experimental results show that MIC greatly speeds up the mining process for a wide variety of datasets. Moreover, MIC obtains most of the high-confidence rules and we show that the missing rules are of little significance using the *interest* measure.

References

[1] IBM Quest Synthetic Data Generation Code. *http://www.almaden.ibm.com/software/quest/*.

[2] R. Agrawal, T. Imielinski, and A. N. Swami. Mining association rules between sets of items in large databases. In *Proc. of SIGMOD*, 1993.

[3] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *Proc. of VLDB*, pages 487–499, 1994.

[4] S. Brin, R. Motwani, and C. Silverstein. Beyond market baskets: Generalizing association rules to correlations. In *Proc. of SIGMOD*, pages 265–276, 1997.

[5] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley & Sons, Inc., 1991.

[6] S. Hettich, C. Blake, and C. Merz. UCI repository of machine learning databases, 1998.

[7] R. Srikant and R. Agrawal. Mining quantitative association rules in large relational tables. In *Proc. of SIGMOD*, 1996.

[8] K. Wang, S. H. W. Tay, and B. Liu. Interestingness-based interval merger for numeric association rules. In *KDD*, pages 121–128, 1998.

[9] M. J. Zaki and K. Gouda. Fast vertical mining using diffsets. In *KDD*, pages 326–335, 2003.

[10] Z. Zhang, Y. Lu, and B. Zhang. An effective partitioning-combining algorithm for discovering quantitative association rules. In *Proc. of PAKDD*, pages 241–251, 1997.