Using Genetic Algorithm for Extracting Association Rules

Dr. Bushra Khireibut Jassim, the University of Baghdad
Afrah Mahmood Abdulla, AL-Mustansiriya University
Dr. Ghassan H. Majeed, the University of Baghdad

Abstract

The process of extracting interesting and unknown information from large database is called as association rule technology. The typical approach for solving association rule problem is Apriori Algorithm developed by Agrawal et al.[1993]. Most of the related existed works are improvements to this algorithm. The limitations of these algorithms are: (1) they required high storage space for saving the huge data resulting the generation of the frequent itemset, (2) they required encoding scheme where separate symbols are used for each possible value of an attribute of the itemset.

In the present work, another trend of solution is proposed. First, we use Genetic Algorithm (GA) to define the maximal frequent itemset, so no huge storage requirement is needed. Also, we force the (GA) to work directly on database, so no encoding scheme is required. The calculations are based on our suggestion to use the variable length individual in the population.

The results obtained in this study demonstrate that the proposed algorithm is a practical one for generation of association rules.

الخلاصة

تدعى عملية أستخلاص المعلومات المهمة والمجهولة من قاعدة بيانات كبيرة بتكنولوجيا القواعد المترابطة. الطريقة النموذجية لايجاد القواعد المترابطة هي خوارزمية Apriori التي أقترحت من قبل Agrawal وجماعته عام 1993. أغلب الاعمال اللاحقة هي تطوير لهذه الخوارزمية. محددات هذه الخوارزميات هي: أنها تحتاج الى مساحة خزنية كبيرة وكذلك أحتياجها الى طريقة للترميز حيث أن كل صفة من صفات مجموعة العناصر تحتاج الى رمز خاص بها.

في العمل الحالي تم أقتراح أسلوب أخر للحل ، في البدء أستخدمنا الخوارزمية الجينية لتحديد مجموعة العناصر المتكررة وبالتالي لم نحتاج الى مساحة خزنية كبيرة، وكذلك وجهنا الخوارزمية للعمل مباشرة على قاعدة البيانات من دون الحاجة الى طريقة للترميز. النتائج المستخلصة من الدراسة الحالية بينت بأن الخوارزمية المقترحة تعطي نتائج عملية لايجاد القواعد المترابطة.

Introduction

Association rule is one of the most important and useful technologies in data mining applications. Association rule technologies extract unknown information from large database and summarize meaningful relation among items to help business managers make better decision. Currently, most of the technologies are focused on basket analysis in supermarkets. Given a set of items and a collection of sales records, which consist in a transaction date and the items bought in the transaction, the task is to find relationships between the items contained in the transactions. In a more general context, where instances are described according several attributes, an association rule is an expression of the form: IF C THEN P.

The IF part is called the rule condition (C) and the THEN part is called the rule prediction (P). Both parts contain a conjunction of terms indicating specific values for specific attributes.

The most famous algorithm to solve association rules is based on the support (frequency of the rule) and the confidence (truth of the rule). Frequent rules are not necessarily interesting and rare rules for which the confidence is very high are also interesting.

In the following example of a bookstore sales database, the association rule mining task is exemplified [1]. There are five different items (authors of novels that the bookstore deals in), I = {A, C, D, T, W}, see Table 1.

Table 1 : Items

Item	Abbreviation
John Ayo	A
Alfred Chihoma	C
Bernard Dungu	D
Thomas Babatunde	T
Peter Walwasa	W

There are six customers in the database who purchased books by these authors. The tables below show the database transactions, all frequent itemsets containing at least three authors (i.e. minimum support = 50 %.) and the association rules.

Table 2 : Database Transactions

Transaction	Items
1	ACTW
2	CDW
3	ACTW
4	ACDW
5	ACDTW
6	CDT

Table 3 : Frequent Itemsets

Support	Itemsets
100%	C
83%	W,CW
67%	A,D,T,AC,A,CD,CT,ACW
50%	AT,DW,TW,ACT,ATW,CDW,CTW,ACTW

Table 4 : Association Rules

A → C(4/4)	AC → W(4/4)	TW → C(3/3)
A → W(4/4)	AT → C(3/3)	AT→ CW(3/3)
A → CW(4/4)	AT → W(3/3)	TW→ AC(3/3)
D → C(4/4)	AW → C(4/4)	ACT→ W(3/3)
T → C(4/4)	DW → C(3/4)	ATW→ C(3/3)
W → C(5/5)	TW → A(3/3)	CTW→ A(3/3)

2. Related Work

The Association Rule mining problem was introduced by Agrawal et al. [2]. They developed the Apriori algorithm for solving the association rule mining problem. Most of the available association rule mining algorithms represent improvements to the Apriori algorithm (Ghosh and Nath [3]; Zhao and Bhowmick [4]), and are called Apriori-based algorithms.

These algorithms work on a binary database, termed as the market basket database. On preparing the market basket database, every record of the original database is represented as a binary record where the fields are defined by a unique value of each attribute in the original database. The fields of this binary database are often termed as an item. For a database having a huge number of attributes and each attribute containing a lot of distinct values, the total number of items will be very large. Storage requirements resulting from the binary database is enormous and as such it is considered one of the weaknesses of the available algorithms.

The Apriori-based algorithms work in two phases. The first phase is for generation frequent itemsets, which are used in the second phase for generating interesting rules. A rule is said to be interesting if its confidence is greater than the user's specified minimum confidence. Frequent itemsets are generated by searching from all-possible item sets whose support is greater than the user specified minimum support. If the value of minimum support is too high, the number of frequent itemsets generated will be small, and thereby resulting in generation of only few rules.

And, if the value is too small, then almost all possible item sets will become frequent and thus a large number of rules may be generated. This causes inference basing on these rules to be difficult. After detecting the frequent item sets in the first phase, the second phase generates the rules using minimum confidence.

Another limitation of the Apriori-based algorithms is the encoding scheme where separate symbols are used for each possible value of an attribute [Ghosh and Nath 2004]. Selecting better rules from them may be another problem.

After detecting the frequent item-sets in the first phase, the second phase generates the rules using another user-defined parameter called minimum confidence (which again affects the generation of rules).

More the number of occurrences better is the rule. The generated rule may have a large number of attributes involved in the rule thereby making it difficult to understand. If the generated rules are not understandable to the user, the user will never use them. Again, since more importance is given to those rules, satisfying number of records, these algorithms may extract some rules from the data that can be easily predicted by the user. It would have been better for the user, if the algorithms can generate some of those rules that are actually hidden inside the data. These algorithms do not give any importance towards the rare events, i.e., interesting rules.

In this paper, we use GA to find maximal frequent itemsets to significantly avoid a large number of redundant works in enumerating relatively short frequent itemset. Since the GA works directly on the data then the need for encoding scheme is eliminated. In the present work it is suggest using variable length individual in the population to avoid the spare problem that existed in the apriori algorithm.

3. Genetic Algorithm for Association Rules (AR)

In this paper we use genetic algorithm(GA) in the difficult part of the (AR) discovery, the part where the frequent itemsets are discovered. Two things must be determined in order to apply a genetic algorithm to a given problem:

i. The genetic code representation

ii. The fitness or objective function, which assigns a quality measure to each solution according to its performance.

The encoding of the parameters in genetic algorithms depends on the problems at hand. A group of individuals (called a population) is stored and modified with each iteration of the algorithm. In (GA), the iterations are referred to as generations. The selection of these individuals is based on their fitness. Individuals in each new generation carry forward genes from the previous generations, and more fit will tend to survive and reproduce.

Individual representation each individual in the population represent a set of items of the transactions in the database.

3.1 Initial Population

The initial population is generated randomly from the set of items in the data base, the number of items in each individual is also chosen randomly.

3.2 Genetic algorithm Operators

- **Selection operator**

The selection of individuals in population will be according to the fitness of individuals by sorting the population

- **Crossover operation**

Two selected individuals crossed over with a point are chosen randomly in each individual according to crossover probability as shown below:

Parent	Offspring
L1 | L2 L3	L1 L6
L4 L5 | L6	L4 L5 L2 L3

- **Mutation Operator**

The chromosome at one position mutates to one item in the item set randomly according to the probability of mutation as shown below (the underline gene):

Before	After
L1 L2 L3	L5 L2 L3

3.3 Fitness Function

As the (GA) used to find maximal frequent itemset, the fitness function must guide the algorithm to these space so the fitness function found by reward the long chromosome with determined weight and penalize the short chromosomes with another weight. Since we preferred not to discard the short chromosome from the population as we need the knowledge found in these chromosomes and also it is difficult to know the long of the maximal frequent itemset found in the data. The maximal frequent itemset differs from one database to another.

4. Algorithm

1. Let itemset be the set of items in which itemset={**t1,t2,……..tn**}
2. Let d (task relevant data) be a set of database transaction where each transactions T is a set of items .
3. Initiate the initial population of GA from the itemset
4. Apply the GA to find the maximal frequent itemsets I.
5. For every maximal frequent itemset, find all nonempty subsets.
6. For every such subset an output a rule of the form a→(I-a), when the ratio of support (I) to support (a) is at least equals to minimum confidence .

5. Case Study

The algorithm is evaluated on a real data that represents the result of students in a certain department of one of Iraqi collages. These results are used to investigate the association between the courses and the effect of each course on another.

We mentioned that the algorithm does not need any encoding schema since the algorithm works directly on the data set which contains the results of 108 students (108 transactions). Since each student has 7 courses, then the itemset will contain 7 items in each transaction.

Some of these transactions are listed below which represent the failure of each student. (The seven courses are denoted by L1 to L7).

Transaction set	
L1 L2 L5 L6	student1 fails in four courses
L1 L2 L3 L4 L5 L6 L7	student2 fails in all courses
L1 L2 L4 L5 L6	
L5	
L1 L2 L3 L4 L5 L6 L7	
L1 L2 L4 L5 L6	
L1 L4 L5	
L1	
L1 L2 L3 L4 L5 L6 L7	
L1 L2 L3 L4 L5 L6	
L1 L2	
L1 L2 L3 L4 L5 L6 L7	
L1 L5	
L1 L5 L6	
L1 L2 L5	
L1 L6	
L1 L2 L3 L4 L5 L6 L7	
L3 L6 L7	
L1 L2 L5 L6	
L1 L2 L3 L4 L5 L6 L7	
L2 L6	
L1 L2 L3 L4 L5 L6	
L1 L2 L3 L4 L5 L6	

Using of GA gives the maximal frequent itemset. Table 4 shows the results for only ten chromosomes.

Table 4

Frequent itemset	Length	Fitness
L6L2L4L5	4	24
L2L5L6	3	30
L6L4	2	43
L2L4L6	3	28
L4L5L2	3	27
L5L4L2L6L3	5	16
L2L6	2	37
L5	1	74
L3L5L6	3	23
L6	1	68

Then we find all subsets, for the best chromosome in the population, that have length more than one, for example L2 L4 L5 L6 has fitness equals to 24.

The fitness represents the support of the maximal frequent itemset whereas the subset of the best chromosome is :{L2}, {L4}, {L5}, {L6}, {L2,L4}, {L2,L5}, {L2,L6}, {L4,L5}, {L4,L6}, {L5,L6}, {L2,L4,L5}, {L2,L4,L6}, {L2.L5.L6}, {L4,L5,L6} for every such subset, the rule and its confidence is calculated. For example

Course L2 with courses L4L5L6 (together), the confidence is (24/46) = 52%

L4→ L2L5L6 , the confidence is (24/58) = 41%

L2L4→L5L6 , the confidence is (24/31) = 77%

L5L6→L2L4 , the confidence is (24/62) = 39%

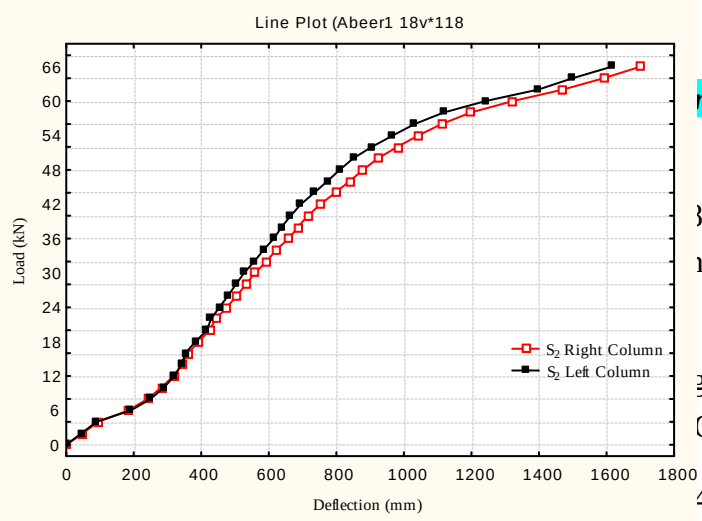
L2L5L6→L4 , the confidence is (24/30) = 80%

Any rule can be selected as an association one if it has a confidence value greater than the minimum one. For our case study, we assume that the minimum confidence is 60%, Based on the above results, course L4 has no association rule with courses L2L5L6. In other words L4 has no effect on courses L2L5L6, so when the student fails in L4 then there is no probability that he will fail in L2 , L5 and L6 courses. However, the results show that failure in courses L2, L5 and L6 will lead to failure in L4 with very high probability. The relation between L2L4 and L5L6 reveals that failing in L2L4 will result in failing in L5L6. But failing in L5L6 will not result in failing in L2L4 (since it is not an association rule).

The above discussion leads to a very useful conclusion. To improve the performance of the students, the head of the department will select an expert teacher to teach the courses that have a strong effect on other courses (that is a strong association rule) or he decides to separate these courses and distribute them into different stages (not in the same stage).

6. Conclusion

- A modified algorithm for finding association rules has been presented which overcomes the limitations of Apriori based algorithms
- The modified algorithm has been applied to a real data base and proved to be easy and straightforward.
- The results of implementation of present study can lead to very useful decisions regarding improving the performance of students, since they discover the dependencies between courses educated.



3 : Extraction of Interesting Association Rules Using
ıal Journal of Computing and ICT Reasearch, Vol.2

g Association Rules Between Sets of Items in Large
CM SIGMOD Conf. on Management of data.

4 : Multi-Objective Rule Mining Using Genetic Algorithms. Information Sciences 163, pp. 123-133.

4. Zhao O. and Bhowmick S. 2003 : Association Rule Mining: A survey. Technical Report, CAIS, Nanyang Technological University, Singapore, No. 2003116.