Association Rule Mining Using Particle Swarm Optimization

Poonam Sehrawat
Dept. of Computer Science
Banasthali University, Banasthali
poonamsehrawat6@gmail.com

Manju
Department of Computer Engg.
CDL Govt. Polytechnic Education Society
Nathusari Chopta, Sirsa
manju.rohil@gmail.com

Harish Rohil
Department of Computer Science& Applications
Chaudhary Devi Lal University, Sirsa
harishrohil@gmail.com

Abstract: Data mining is the process of discovering new relevant information in terms of patterns from large amount of data. Association rule mining is one of very important data mining techniques. Swarm optimization is a new subfield of artificial intelligence which studies the cooperative performance of simple agents. In this paper, proposed a new efficient algorithm for exploring high-class association rules by particle swarm optimization (PSO) algorithm. The proposed method mine interesting and understandable association rules without using the minimum support and the minimum confidence thresholds in only single scan. To prove the practical significance of the approach, this approach is implemented on Microsoft Visual Studio 4.0. Experimental evaluation shows the efficiency of proposed algorithm in terms of computation time.

Keywords: Data Mining, Association Rule Mining, Particle Swarm Optimization.

1. Introduction

With the development of information technology, there are many different kinds of information databases, such as scientific data, medical data, financial data, and marketing transaction data. How to effectively analyze and apply these data and find the critical hidden information from these databases have become very important issues. Data mining has attracted a great deal of
attention in the information industry, scientific analysis, business application, medical research and in society as a whole in recent years due to the wide availability of huge amounts of data [11].

Data mining can be categorized into several models, including association rules, clustering and classification. Among these models, association rule mining is the most widely significant method. The task of association rule mining in large database to find out the frequent patterns or itemsets and discover association rule corresponding their frequent itemsets. The following is a formal statement of the problem [1]: Consider I= \{i_1, i_2, \ldots, i_m\} be a set of literals, called items. D be a set of transactions, where each transaction T is a set of items such that T \subseteq I. Associated with each transaction is a unique identifier, called its TID. We say that a transaction T contains X, a set of some items in I, if X \subseteq T. An association rule is an implication of the form X \Rightarrow Y, where X \subseteq I, Y \subseteq I, and X \cap Y = \emptyset and indicates that the presence of items X in the transaction as well as presence of items Y. However, the mining of association rule depend upon two parameters [1]:

(1) **Minimal support**: The support of rule X\Rightarrow Y is the fraction of transactions in D containing both X and Y and finding frequent itemsets with their supports above the minimal support threshold.

\[
\text{Support} (X \Rightarrow Y) = \frac{|X \cup Y|}{|D|}
\]  

(2) **Minimal confidence**: The confidence of rule X\Rightarrow Y is the fraction of transactions in D containing X that also contain Y and generate association rules that have confidence above the minimal confidence threshold.

\[
\text{Confidence} (X \Rightarrow Y) = \frac{\text{Sup}(X \cup Y)}{\text{Sup}(X)}
\]

The association rule mining problem divide into two subproblems [1]:

(1) The discovery of frequent itemsets.
The generation of association or desired rules.
The Aprior algorithm is an influential algorithm for association rule mining in large database. This algorithm uses previous knowledge of frequent itemsets. It repeatedly generates candidate itemsets from previous level of candidate itemsets and uses minimal support and confidence to prune these candidate itemsets to find frequent itemsets.

Particle swarm optimization (PSO) is a population based method, where a population is called a swarm. In PSO, each particle has potential solution or randomized velocity flown through problem space. Each particle update their velocity and position according to previous position or its own best position which is found so far and global best position which is overall best value in swarm.

2. Related Work

2.1. Association Rule Mining Algorithm

Association rule mining, one of the important and well researched techniques of data mining, used to find out the important correlations among data items in the database and some hidden relationships exist between purchased items in transactional databases was first proposed by Agrawal in 1993. Support and Confidence are two important interestingness measures. It aims to find out to extract frequent or interesting patterns or association between set of items in the database.

The Aprior algorithm (Agrawal, Imielinski & swami, 1993; Agrawal & Srikant, 1994; Agrawal & Shafer, 1996) is the most representative algorithm of association rule mining. So far, many algorithms proposed for discovering association rules are Apriori [1], SETM [2], AIS [1] and other methods. However, these algorithms have their limitations. They have to mine association rules in two stages separately. In these methods, rules with high occurrence in database are considered as the best rules, whereas most of these rules can easily be predicted by the users. Therefore, they are not interesting. Also, they mine occurrence rules with a large number of attributes, which are not understandable for the user. Hence, the user will never use them. In these methods, two parameters, minimum support and minimum confidence thresholds, are always determined by the decision-maker or through trial-and-error; and hence, these algorithms lack both objectiveness and efficiency [2].
In order to improve the efficiency of Apriori, many researchers have proposed modified association rule-related algorithms. Park et al. [5] introduced a hash based DHP technique improving the efficiency of aprior algorithm by reducing the size of candidate k-itemsets, $C_k$ for $k>1$. DHP includes two major features, the efficient generation of large itemsets and the effective reduction of transaction database sizes.

The parallel algorithms for the discovery of association rules using clustering techniques to approximate the set of potentially maximal frequent itemsets was first proposed by Zaki et al in 1997. This algorithm uses two clustering schemes based on equivalence classes and maximal hypergraph cliques, and study two lattice traversal techniques based on bottom-up and hybrid search and also use vertical database layout to cluster related transactions together [6].

### 2.2 Particle Swarm Optimization Algorithm

Kennedy and Eberhart proposed particle swarm optimization concept in 1995 is a population based meta-heuristic algorithm. The PSO algorithm simulates the behaviors of bird flocking and fish schooling. PSO is popular because of its social behavior. Each single solution is a “bird” in the search space and these single solutions considered as a “particle” in the search space. All particles have fitness values, which are evaluated by the fitness function to be optimized. The particles also have velocities which direct the flight of the particles.

Each particle has random velocity and adjusts it according to its own flying experience and its neighborhood experience. Each particle is updated by the two best values in every iteration. Each particle keeps track of its coordinates in the problem space which are associated with the best solution (fitness) it has achieved so far. (The fitness value is also stored.) This value is called pbest. Another "best" value that is tracked by the particle swarm optimizer is the best value, obtained so far by any particle in the swarm. This best value is a global best and is called gbest.

After finding the two best values, each particle updates its velocity and position according to Eqs. (3) and (4) [7]:

$$v_{id}(t+1) = v_{id}(t) + c_1 \text{ rand}(pbest-x_{id}(t)) + c_2 \text{ Rand}(gbest-x_{id}(t))$$  \hspace{1cm} (3)

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1)$$  \hspace{1cm} (4)
Where \( v_{id}(t+1) \) is the velocity of the \( i^{th} \) particle at time \( (t+1) \) in \( D \)-dimensional space and \( v_{id}(t) \) is the velocity of the \( i^{th} \) particle at time \( t \). \( x_{id} \) is the current particle position. \( \text{rand}() \) and \( \text{Rand}() \) are random numbers in \((0, 1)\); \( c_1 \) is the cognitive factor; \( c_2 \) is the social factor. Usually \( c_1 \) and \( c_2 \) are set to be 2. The velocities of particles in each dimension are clamped to a maximum velocity \( V_{\text{max}} \geq V \geq -V_{\text{max}} \).

Kennedy and Eberhart proposed discrete binary particle swarm optimization extends the capability of the continuous PSO [8].

In 1998, Shi and Eberhart proposed another method called the “Linearly decreasing inertia weight method.” In this method, the particle updates its velocity and position with Eqs. (5) and (6) as follows [9]:

\[
v_{id}(t+1) = \omega v_{id}(t) + c_1 \text{rand}(pbest-x_{id}(t)) + c_2 \text{Rand}(gbest-x_{id}(t)) \quad (5)
\]

\[
x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (6)
\]

Clerc and Kennedy proposed “Constricted PSO” to improve the convergence rate. They used Constriction factor (\( K \)) to prevent explosion and particles to convergence on a local optima. In Clerc’s method, the particle updates its velocity and position with Eqs. (7) and (8) [10]:

\[
v_{id}(t+1) = \chi[v_{id}(t) + c_1 \text{rand}(pbest-x_{id}(t)) + c_2 \text{Rand}(gbest-x_{id}(t))] \quad (7)
\]

\[
x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (8)
\]

\[
\chi = \frac{2}{2-\varphi - \sqrt{\varphi^2 - 4\varphi}} \quad \text{where } \varphi = c_1 + c_2, \varphi > 4
\]

\[
(9)
\]

Fully Informed particle swarm (FIPS uses information from its entire neighbor rather than the best one [13].
Kuo et al proposed propose a novel algorithm for association rule mining in order to improve computational efficiency as well as to automatically determine suitable threshold values. The particle swarm optimization algorithm first searches for the optimum fitness value of each particle and then finds corresponding support and confidence as minimal threshold values after the data are transformed into binary values and then these minimal support and confidence values are used to mine association rules [12].

3. Proposed Association Rule Mining Using Particle Swarm Optimization (ARMPSEO)

In this work, particle swarm optimization algorithm which is one of the new evolutionary algorithms is applied to explore association rules from transactional databases. This approach is called ARMPSEO. The following of this section are some important parts of the algorithm which are explained: particles encoding, fitness function, and finally the last part of the section explain the pseudo-code of ARMPSEO.

3.1. Particles Encoding

In this paper, each particle represents a rule and each rule contains of a series of decision variables which represent the status of every item in the rule. According to Figure 1 in the proposed algorithm, every particle has n decision variables in lieu of n items in any dataset.

![Figure 1: Particles Encoding](image)

This means that the $i$th variable which is known as $ES_i$ indicates the status of $i$th item and can take values between 0 and 1. In this way, if $0.00 \leq ES_i \leq 0.33$, the $i$th attribute is in the antecedent of the rule and if $0.33 \leq ES_i \leq 0.66$, this attribute is in the consequence of the rule and if $0.66 \leq ES_i \leq 1$, it means the lack of $i$th attribute in the rule.

3.2. Fitness Function

The fitness function provided in this study is in (10)
fit(i) = \alpha_1 \left[ \frac{\text{Sup}(A \cup C)}{\text{Sup}(A)} \right] \cdot \left[ \frac{\text{Sup}(A \cup C)}{\text{Sup}(C)} \right] \cdot \left[ 1 - \frac{\text{Sup}(A \cup C)}{|D|} \right] + \alpha_2 \frac{\text{NumberField}(i)}{\text{MaxField}} \tag{10}

Since the mining association rule is a task that extracts some hidden information, it must discover those rules that have a comparatively less occurrence in the entire database which are more interesting for the users; discovering such rules is more difficult. For classification rules it can be defined by information gain theoretic measures. But it is not efficient for evaluating the association rules. Therefore, interestingness measure in [15] is used in the fitness function according to the first parameter. In this parameter |D| is the total number of records in the database. This relation has three parts:

1. \left[ \frac{\text{Sup}(A \cup C)}{\text{Sup}(A)} \right] shows the probability of creating the rule depending on the antecedent part;

2. \left[ \frac{\text{Sup}(A \cup C)}{\text{Sup}(C)} \right] shows the probability of creating the rule depending on the consequent part.

In fact most of these are interesting rules in which the rate of acquired information is approximately the same in both antecedent and consequent parts of the rule. In this parameter the support count of the rule antecedent and the support count of the rule consequent are used.

3. \left[ \frac{\text{Sup}(A \cup C)}{|D|} \right] gives the probability of generating the rule depending on the whole data-set.

So complement of this probability will be the probability of not generating the rule. Thus, a rule having a very high support count will be measured as less interesting, because such rules easily predictable by user.

The second parameter is used for number of attributes in rule and it rewards the shorter rules with a smaller number of attributes. Numberfield(i) returns the number of attributes that exist in particle i. This term is to lead to shorter rules. In result, comprehensibility of rules that are important in data mining is increased. Larger rules are more likely to contain redundant information.

It should be noted that \alpha_1 and \alpha_2 will be specified by the percent of user interests and one might increase or decrease the effects of parameters of the fitness function.

3.3. Proposed ARMPSO Approach

The proposed approach contains two parts:
1) The first part provides procedures related to encoding and calculating the fitness values of each particle in swarm.

2) In the second part of the approach, which is the main contribution of this study, the particle swarm optimization algorithm is employed to mine the association rules.

The pseudo code of proposed ARMPSO approach shown in Figure 2, the algorithm is run as N times or number of desired rules. Moreover, each run includes a number of generations.

```
Input: Database of Transaction (D)
Output: Best discovered association rules

1. t=0     // here t is iteration number
2. DiscoveredBestRules=Ø
3. initialize a population (M) particles
4. While(t<N) do   // algorithm run for number of iterations(N)
5. begin
6. i=0     // here loop starts for each particle
7. repeat
8. calculate fitness of each particle
9. For each particles P do
10. begin
11. For each dimension d in P
12. begin
13. calculate velocity and position of each particle in d-dimension
14. end for each dimension
15. end for each particle
16. i++
17. Best ← Particle [Bestparticleindex]
18. until not terminate(i)
19. DiscoveredBestRules ← DiscoveredBestRules U Best
20. t++
21. End
```

**Figure 2: Pseudo-Code of Proposed Association Rule Mining Using Particle Swarm Optimization**
At the beginning of the algorithm, the DiscoveredBestRules set is empty. At the first iteration of algorithm, each particle is initialized randomly as a rule. In each of generation, until the reaching the termination conditions, particles are evaluated and then calculate fitness of each particles. The individuals of population are sorted in descending order according to their fitness value. In each generation, the best discovered rule is added to Best vector and then for each particles update their velocity and position according to Eqs. (3) and (4). The rule is valid if it has at least one attribute in the antecedent of the rule and one in the consequence of the rule. At each iteration after reaching the termination condition best discovered rule from the Best is added to the DiscoveredBestRules set. This process is continued until termination conditions not occur.

4. Experimental Evaluation

A simulator with GUI was designed and developed with Microsoft Visual Basic 4.0 using C# and run on a 2.10 GHz machine with a RAM of 2GB. This simulator accepting input in the form of text file and output of proposed approach is also text file which show the best discovered rules. Assessment of proposed approach was made on the basis of: Execution time refers most literally to the time during which a computer program is executing. In this research work, execution time is taken with respect to population size. This experiment evaluates the efficiency and usefulness of the proposed algorithm. For conducting the examination, the authors have used the dataset; format of dataset is taken from the Iris dataset. Iris dataset is online available at http://www.ics.uci.edu/~mlearn. The specifications of dataset are already given in Table 1

<table>
<thead>
<tr>
<th>Dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. record</td>
</tr>
<tr>
<td>No. attribute</td>
</tr>
</tbody>
</table>
There are total 42 rules are extract out of 50 transaction database. ARMPPO algorithm runs for 10 iterations and 5 Best Discovered Rules are generates as shown in Table 2. Best discovered rules are changed in each run because each time algorithm uses different random value.

Table 2: Best Discovered Rules of ARMPPO Approach

<table>
<thead>
<tr>
<th>Iteration Number</th>
<th>Best Discovered Rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SOE⇒B</td>
</tr>
<tr>
<td>2</td>
<td>SOE⇒B</td>
</tr>
<tr>
<td>3</td>
<td>LI ⇒RB</td>
</tr>
<tr>
<td>4</td>
<td>LI ⇒RB</td>
</tr>
<tr>
<td>5</td>
<td>OFA ⇒Q</td>
</tr>
<tr>
<td>6</td>
<td>SOE⇒B</td>
</tr>
<tr>
<td>7</td>
<td>SOE⇒B</td>
</tr>
<tr>
<td>8</td>
<td>SOE⇒B</td>
</tr>
<tr>
<td>9</td>
<td>LI ⇒RB</td>
</tr>
<tr>
<td>10</td>
<td>LI ⇒RB</td>
</tr>
</tbody>
</table>

Association rule mining using particle swarm optimization is efficient in terms of execution times with respect to population size than the association rule mining using firefly algorithm (ARMFA) [14] as shown in Figure 3. There is a little variation in run time of both proposed approach clearly shown from Figure 3. The proposed approach and ARMFA approach started from population size 5, at population size 5 run times of both approaches are same. After that population size proposed ARMPSO approach take less time as compared with existing similar approach. It is obvious that the ARMPSO is better than the ARMFA and automatically better than the existing approach proposed by Kuo et al [12].
Figure 3: Relationships between Population Size and Computation Time for ARMPSO and ARMFA Approach

Figure 3 clearly indicates that the proposed ARMPSO algorithm outperforms the ARMFA between population size and computation time.

5. Conclusion

The work reported in this paper presents an efficient approach for exploring high quality association rule. The proposed approach is based on particle swarm optimization. In this proposed approach, encoding of particles is applied to extract rules from database. For extracting rule, fitness value of individual rule is computed instead of minimum support and minimum confidence thresholds. This has an advantage that the database is scanned once only which enhances efficiency of the approach in terms of CPU time and memory consumption. The proposed approach was implemented using Microsoft Visual Studio 4.0 and the result was compared with existing similar approach proposed by Poonam et al. The proposed ARMPSO algorithm betters the application of particle swarm optimization in terms of relationship between population size and computation time.

References


