New Binary PSO based Method for finding best thresholds in association rule mining

Abdoljabbar Asadi[1], Mehdi Afzali[2] ,Azad Shojaei [*3], Sadegh Sulaimani[4]

[1] Department of Computer Engineering, Zanjan Branch, Islamic Azad University, Zanjan, Ira
[2] Department of Information Technology Engineering, Zanjan Branch, Islamic Azad University, Zanjan, Iran
[3 *]Department of Computer, Saghez Branch, Islamic Azad University, Saghez, Iran
[4] IT and Computer Eng. Dept., University of Kurdistan, Sanandaj, Iran
Azad.Shojaei@gmail.com

Abstract: One of the important data mining techniques is association rule finding. Apriori is the most famous algorithm based on this technique. But it has a major weakness which cannot calculate the minimal value of support and confidence and these parameters is estimating intuitively by the user and this has an important effect on the algorithm performance. Main goal of this paper is to presenting an optimal method to find suitable values of minimum threshold for support and confidence by means of Binary Particle Swarm Optimization. Data used for the paper is a 4000 random records sample from Foodmart 2000 Database. Implementation of the proposed method has been done using R2010b version of MATLAB software. Proposed algorithm improves the performance of association rule mining by automatically setting suitable values for minimum support and confidence thresholds.. [Abdoljabbar Asadi, Mehdi Afzali ,Azad Shojaei, Sadegh Sulaimani , **New Binary PSO based Method for finding best thresholds in association rule mining.** *Life Sci J* 2012;9(4):260-264] (ISSN:1097-8135). http://www.lifesciencesite.com. 37

Keywords: data mining, association rule mining, minimal support, minimal confidence, particle swarm optimization

1. Introduction

Regard to information technology development many databases have been developed to store the related data. Analyses of these databases to mine hidden rules have incremental importance [1]. One of the noticeable techniques help managers to make good decisions is data mining. This technique makes great tools available for users during current decay to extract meaningful information and useful patterns from databases [2], and this knowledge should be exact, readable and easy to understand [3]. But in spite of vast area of applications for data mining it still needs some manual operations, not automatic, to complete [3]. One of the important data mining techniques is association rule finding. It can extract hidden rules and dependent properties which have important role in decision making [1]. Apriorri Algorithm is the most famous one to extract association rule mining. But this algorithm has a major weakness which cannot calculate the minimal value of support and confidence and these parameters is estimating intuitively [3]. There are several algorithms to improve performance and accuracy of Apriorri. In traditional algorithms of association rule mining both of support and confidence parameters minimal value is chosen by the user try and error and this has an important effect on algorithm performance [1]. This approach can also produce many rules in a large database, millions, which probably many of them are not useful; it can be implied that it doesn't have enough efficiency [20]. So we need a method to find best values of support and confidence parameters automatically specially in large databases. Main goal of this paper is to presenting an optimal method to find suitable values of minimum threshold for support and confidence efficiently. This aim is achieved using particle swarm optimization (PSO) algorithm. PSO as an optimization method [5, 6] can be used for optimization of association rule mining [4]. A special type of PSO, named Binary PSO, is used for our work regard to its efficiency for local and large interval domains [21].

2. Literature Review

Because of long runtime of Apriori algorithm to find association rules, its operational efficiency has a considerable importance. Several papers have presented different association rule mining algorithms to improve Apriori algorithm. Savasere et al. [8] developed Partition Algorithm for association rule discovery which is basically different form classic algorithm. This algorithm first scans the database to find strong item sets. Then support value is calculated for all item sets. Validity hint of Partition algorithm is that any strong item set appear in a section at least one time. Park et al. [22] introduced DHP at 1995. DHP is a derivative of Apriori plus some extra controls. It uses hash table to restrict candidates. DHP has two main properties: effective make of item sets and efficient reduction of database size by dropping adverse attributes. Toivonen et al. [23] presented sampling algorithm at 1996. This algorithm is about finding association rules according to reduce database operations. DIC algorithm by Brin et al. at 1997 [9] splits database into some parts called start point. It determines support value for item sets belong to each start point and so extracts the patterns and rules. Bender search algorithm [10] by Lin et al. developed at 1998 can discover rules from most frequent itemsets. Yang et al. offered an efficient hash based method named HMFS which combines DHP and Bender search algorithms results in reduction of database scan and filtering repeated itemsets to find greatest repeated itemset [11]. This can shorten overall computation time for finding greatest repeated itemset. Genetics algorithm has been applied to association rule mining during recent years. [12] Utilizes weighted items to distinguish unique itemsets. Value of different rules is determined using weighted items in fitness function. This algorithm can find suitable threshold value for association rule mining. Saggar [13] et al. presented a method for optimizing extracted rules, using genetics algorithm. The importance of the work is that it can predict rules with negative value.

Kuo et al. [1] in 2011 developed a PSO based method for automatic finding threshold value of minimal support. Their work shows that basic PSO can find values faster and better than genetics algorithm. Gupta [3] also offered a method at 2011 for automatic finding of threshold value using weighted PSO. His results show high efficiency of PSO for associative rule mining. This approach also can gain better values of threshold in comparison with previous ones.

3. Basics

3.1 Association rule mining

Agrawal et al. raised associative rule mining idea at 1993 [14]. A positive association rule presented as *if* $A \rightarrow B$ which *A* and *B* are subsets of *itemset(I)* and each itemset includes all of the items $\{i_1, i_2, ..., i_n\}$; It can be shown that in database $D = \{T1, T2, . . ., Tk\}$ a customer buys *B* product after buying *A* one if $A \cap B \neq \emptyset$. Association rule mining should be based on the following two parameters:

1. Minimum support: finding item sets with the value above threshold

$$Support(A \rightarrow B) = P(A \cup B) = \frac{A \cup B}{D} \quad (1)$$

2. Minimum Confidence: finding item sets with the value above threshold

$$Confidence(A \rightarrow B) = p(B|A) = \frac{A \cup B}{A} \quad (2)$$

Better rules have greater support and confidence value. Most famous algorithm for association rule mining is Apriori, offered by Agrawal et al. It repeatedly determines candidate itemsets using minimal support and confidence to filter itemsets for finding repeated ones with more frequency [1].

3.2 Particle Swarm Optimization Algorithm

PSO algorithm first developed at 1995 by James Kennedy, Russell C. Eberhart. It uses a simple mechanism inspiring from simultaneous motion of birds and fishes fly and their social life. This algorithm has successful applications recent years [5, 6]; mainly neural network weighting and control systems and everywhere that genetic algorithms can be use. PSO is not only a tool for optimization but also a tool for human social recognition representation. Some scientists believe that knowledge will optimize in effect of mutual social behaviors and thinking is not only a private action, indeed it is a social one. There are some entities in search space of the function which we are going to optimize it, namely particles [15]. PSO as an optimization algorithm provides a population based search which every particle change its position according to the time. Kendy in 1998 represented that each particle can be a possible answer that can move randomly in problem search space. Position change of each particle in search space is affected by experience and knowledge of itself and its neighbors [16]. Suppose we have a *d* dimension space and *i*'th particle from the swarm can be present with a velocity vector and position vector. Position change of each particle is possible by change in position structure and previous velocity. Position of each particle is x_i and it has information about best value which has reached yet, named *pbest*. This information is obtained from particles attempt to reach the best answer. Also any particle knows the best answer obtained for *pbest* from others in the swarm, named *gbest*. Each particle tries to change its position in order to reach the best solution using the following parameters:

x_i current situation, v_i the velocity, destination between the current position and *pbest*, destination between current position and *gbest*.

So the velocity of each particle changes as follows:

$$(pbest_i - x_i^k) + c_2 r_2 \ . \qquad (3)$$

$$(gbest - x_i^k) V_i^{k+1} = w v_i^k + c_1 r_1 \ .$$

Which V_i^k is the velocity of each particle in *k*'th repeat, *w* is the inertia weight, c1 and c2 are learning coefficients, r_1 and r_2 are random variables in the *[0,1)* interval with the unique distribution, x_i position of each particle *i* in the *k*'th repeat, $pbest_i$ which is *pbest* of *i*'th particle and *gbest which is gbest* of the group. Maximum of velocity (V_{max}) is to prevent velocity from increasing unlimitedly [17,18]. Position of each particle is determined as follows:

$$X_i^{k+1} = x_i^k + v_i^{k+1} \quad (4)$$

Equations 1 and 2 are form primitive version of PSO algorithm. PSO algorithm is so easy and has low computational, speed and memory load. It is using to solve continues problems while our work needs discrete version of the PSO. One of the discrete versions is binary PSO which has developed by Kennedy and Eberhart at 1997 [6]. They did a small change on the algorithm to support discrete quantities also. Velocity is used as a probabilistic threshold value here and can be 0 or 1. X_j^i, value of *j'th* bit from binary vector, shows the *i'th* particle position. So the following describes Binary PSO function [7]:

$$X^i_j[t] = \begin{cases} 1 & ,\sigma < s(v[t]) \\ 0 & ,otherwise \end{cases} \quad (5)$$

Which **σ** is a random number with the uniform distribution in [0,1] interval. **s(.)** is also the Sigmoid function described as follows:

$$S(z) = \frac{1}{1+\exp\,(-z)} \qquad (6)$$

Velocity change in Binary PSO is the same way as standard PSO.

4. Method

Research steps of this paper are as the Figure 1. This is composed of two main parts: Data Preprocessing (Steps 1 to 4) and Data Mining (Step 5 and 6). First we describe data collection (Step 2) and preprocess it (Steps 3 to 4) and then we use the algorithm to mine association rules (Steps 5 and 6). In the part one we prepare the database to apply the algorithm. First we preprocess it in order to make it ready to change to binary form. After converting it to the binary form, well form for using with binary PSO, we will generate primary population in order to find *gbest*. Then we apply PSO to find minimum threshold value of support and confidence using particle movement funded optimal values. Having the best threshold we can use it for association rule mining resulting into better results. Hardest part of the research is to collect data and prepare it. Data is the primitive thing used in data mining. So it is important for good data mining to access and implement suitable data [19]. Data used for this paper is from Foodmart 2000 Database in the format of Microsoft SQL Server 2000, *Sales_fact_1997* table [1,3,24].

There are 86837 records from sale activity of a supermarket. Each record contains 8 properties. Our sample includes of 4000 random records. We have used optimal binary PSO to improve positive and negative rule production. Each particle represents a positive rule; consist of a predecessor and a successor. Figure 2 shows a particle; orange color is predecessor and blue one is successor. Every box represents a field from database. Containment of the boxes presents the value of a field in the database in the binary format.

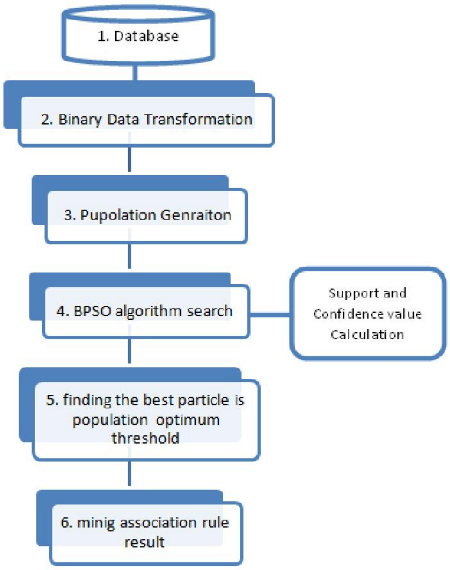
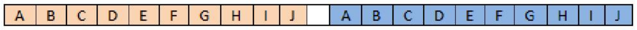

1. Database
2. Binary Data Transformation
3. Pupolation Genraiton
4. BPSO algorithm search
Support and Confidence value Calculation
5. finding the best particle is population optimum threshold
6. minig association rule result

Fig1. Steps of the proposed method

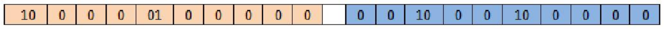


A	B	C	D	E	F	G	H	I	J		A	B	C	D	E	F	G	H	I	J

Fig2. Presentation of a particle

For example Fig. 3 shows a rule with the following specifications:

IF(product_id = 53 AND store_id =1)→(customer_id = 3, store_sales = 3)



10	0	0	0	01	0	0	0	0	0		0	0	10	0	0	10	0	0	0	0

Fig3. Example of a particle in the database

Implementation of the proposed method has been done using *R2010b* version of MATLAB software. Movement representation of the particles toward the best goal is prepared form MATLAB also. Guiding a particle from the swarm population to an optimal answer is done by the fitness function. The particle with the greatest value of fitness usually supposed as the best particle [1] and [3]. In the proposed method *A* and *B* are collections of properties participating at predecessor and successor obtained from decoding respective particle according to what is explained. We calculate support and confidence values as follows:

In order to producing rule in the form of *if A* → *B*, two criteria form *cost(p)*function has been used to evaluate association rules quality.

$$Support = \frac{Supp(A \cup B)}{N} \qquad (7)$$

$$Confidence = \frac{Supp(A \cup B)}{Supp(A)} \qquad (8)$$

After sending the particles to the fitness function, particle with the greatest fitness level will be used to move other particles toward the most optimal rule. Fitness function is defined as follows:

Fitness= α_1**Support* +α_2**Confidenss* –α_3**NA*

Which *NA* is the number of properties used in the rule and coefficients, $\alpha_1,\alpha_2,\alpha_3$, is used to parametric control of fitness function and customized by the user. First and second parts of this function is related to support and confidence values. It is essential to take into account both parts simultaneously. Because only one of support or confidence values cannot be a criteria for quality assessment of produced rules. It is evident that the more the value of both factors simultaneously the better the quality of the rule. We know that long rules will probably result to low quality productions also. So we try to produce relatively short, readable rules with more concept and quality which has special importance in data mining [3].= First *n* particles are creating quite randomly, each one representing a rule. Then fitness value of each one will be evaluated using the function noticed before. Binary PSO search algorithm will run until reaching the end condition; i.e. the best particle has founded and support and confidence value of it can be used as minimal support and minimal confidence. So we can utilize them for mining better and more association rules.

5. Results and Discussion

Running proposed method on the sample noticed before, 4000 random samples from Sales_fact_1997 tables of Foodmart 2000 database, showed satisfactory results. We set the PSO algorithm parameters as follows:

Table 1. PSO Algorithm Parameters

Repeat Numbers	Learning rate of C_1,C_2	α_3	α_2	α_1
7	2	0.2	0.8	0.8

After running the algorithm for five population sizes, Table 2 was obtained:

We compared our results with those obtained from latest reference, reference [3]. Results were so better. Fig. 4 compares confidence acquired by two algorithms and Fig. 5 compares acquired support. Red curve is our proposed algorithm while blue one is from [3]. X axis is the population size.

Obtained results form proposed algorithm will lead to better association rule mining, because of fine tuned minimum support and confidence. So we can conclude that proposed algorithm can improve the performance of association rule mining.

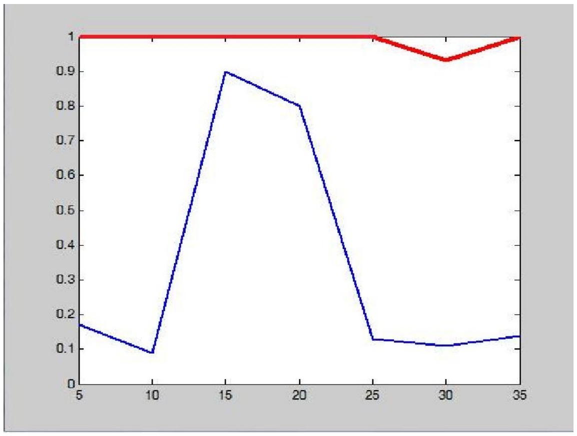


Fig. 4- Confidence compare of two algorithms

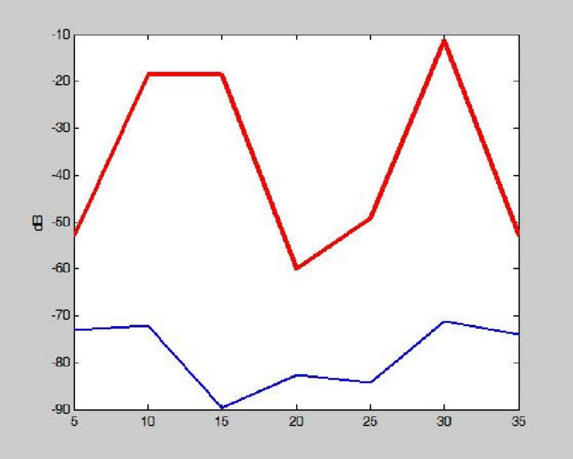


Fig. 5- Support compare of two algorithms

Table 2. Results of the proposed algorithm

Optimal Rule	State	Population Size	number of rules	discovered confidence	Discovered support	Runtime (sec)
a3=9642 --> a5=3	1	5	350	1	0.005	92
a8=1 --> a5=2	2	10	700	1	0.155	129
a8=1 --> a5=2	3	15	1050	1	0.155	220
a4=129 a6=4.9 a8=2 --> a5=2	4	20	1400	1	0.0025	314
a4=188 a8=2 --> a5=3	5	25	1750	1	0.0073	401
a8=3 --> a5=3	6	30	2100	0.93	0.32	471
a4=120--> a5=2	7	35	2450	1	0.005	562

6. Conclusion

Main disadvantage of association rule mining like Apriori is the intellectual computation of minimum support and confidence. Using the binary PSO algorithm, it is possible to compute both parameters quickly and efficiently. Results show improvements in comparison with previous methods. So we propose to add a prior computation to association rule mining before starting the task to determine minimum support and confidence. Future works can focus on improving PSO algorithm speed and considering negative rule mining in addition of positive rules. Because negative rules has the same importance of positive rules for managers. It is possible to suggest new methods to association mining of negative also.

Corresponding Author:

Azad Shojaei
Islamic Azad University, Saghez Branch, Saghez, Iran . E-mail: azad.shojaei@gmail.com

References

[*1*] R.J. Kuoa, C.M. Chaob and Y.T. Chiuc ."Application of particle swarm optimization to association rule mining": Applied Soft Computing 11 (2011) pp:326–336.

[2] Olafsson Sigurdur, Li Xiaonan, and Wu Shuning "Operations research and data mining, in": European Journal of Operational Research 187 (2008) pp:1429–1448.

[3] Manisha Gupta."Application of Weighted Particle Swarm Optimization in Association Rule Mining". International Journal of Computer Science and Informatics (2011) vol.1 pp:69-74.

[4] Maragatham G, Lakshmi M. "A RECENT REVIEW ON ASSOCIATION RULE MINING": Indian Journal of Computer Science and Engineering (2012) Vol. 2 pp:831-836.

[5] j.kennedy and r.c. eberhart." particle swarm optimization": IEEE Int. Conf. Neural Netw. Perth, Australia (1995) vol. 4 pp: 1942-1948.

[6] R. C. Eberhart and J. Kennedy. "A new ptimizer using particle swarm theory". 6th Int. Symp. Micromachine Human Sci., Nagoya, Japan, 1995, pp. 39–43.

[7] Riccardo poli, James Kennedy, Tim Blackwell ."Praticle swarm optimization An overview ":Springer Science.swarm intell(2007) pp:33-57.

[8] A. Savasere, E. Omiecinski, S. Navathe, "An efficient algorithm for mining association rules in large database", in: Proceedings of the 21st VLDB Conference, 1995, pp. 432–444.

[9] H. Toivonen, "Sampling large databases for association rules", in: Proceedings of the 22nd VLDB Conference, 1996, pp. 134–145.

[10] D.I. Lin, Z.M. Kedem, Pincer search:" a new algorithm for discovering the maximum frequent set", in: Proceeding of the 6th International Conference on Extending Database Technology: Advances in Database Technology, 1998, pp.105–119.

[11] D.L. Yang, C.T. Pan, Y.C. Chung, "An efficient hash-based method for discovering the maximal frequent set", in: Proceeding of the 25th Annual International Conference on Computer Software and Applications, 2001, pp. 516–551.

[12] S.S. Gun, "Application of genetic algorithm and weighted itemset for association rule mining", Master Thesis, Department of Industrial Engineering and Management, Yuan-Chi University, 2002.

[13] M. Saggar, A.K. Agrawal, A. Lad, "Optimization of association rule mining using improved genetic algorithms", in: Proceeding of the IEEE International Conference on Systems Man and Cybernetics, vol. 4, 2004, pp. 3725–3729.

[14] R. Agrawal, T. Imielin´ ski, A. Swami. "Mining association rules between sets of items in large databases":ACM SIGMOD Record 22 (2) (1993) pp:207–216.

[15] Riccardo poli, James Kennedy, Tim Blackwell ."Praticle swarm optimization An overview": Springer Science. swarm intell(2007) pp:33-57.

[16] Kennedy, J. "The behavior of particle"s: porto,v.w, Saravanan,N.,Waagen.D.,andEiben,A.E(eds.),In:EvolutionaryProgrammingVII,Springer (1998) pp:581-590.

[17] Y. Shi , R. Eberhart. Parameter selection in particle swarm optimiza-tion: 7th Int. Conf. Evol. Program., NCS (1998) vol. 1447 pp: 591–600.

[18] R. Eberhart , Y. Shi."Comparing inertia weights and constrictionfactors in particle swarm optimization": IEEE Congr. Evol.Comput (2000) pp: 84–88.

[19] Philippe Lenca, Patrick Meyer, Bonoit vaillant, Stephae lallich." On selecting interestingness measures for association rules": User oriented description and multiple criteria decision aid: European Journal of operation research (2008)184 610 – 626.

[20] Veenu Mangat. "Swarm Intelligence Based Technique for Rule Mining in the Medical Domain": International Journal of Computer Applications. Volume 4(2010)pp :19-24.

[21] Kennedy, J., & Eberhart, R. C." A discrete binary version of the particle swarm algorithm". In Proceedings of the conference on systems, man, and cybernetics. (1997).pp: 4104–4109. Piscataway: IEEE.

[22] Park, J. S., Chen, M., & Yu, P. "An effective hash-based algorithm for mining association rules". (1995). Pp: 175–186. International Conference on Management of Data.

[23] Hannu Toivonen. "Sampling large databases for Association Rules"(1996) pp:1-12.VLDB conference. india.

[24]http://social.msdn.microsoft.com/Forums/enUS/sqlanalysisservices/thread/1fbade48-8f92-4eb9-ac65-a01593c5d228

9/3/2012