The Imperialist Competitive Algorithm for Automated Mining of Association Rules

Fariba Khademolghorani
Department of Computer Engineering
Islamic Azad University
Najafabad Branch, Iran
Fariba.Khadem@sco.iaun.ac.ir

Journal of Digital
Information Management

ABSTRACT: *Association rule mining is an optimization problem because of several limitations. Recently, the imperialist competitive algorithm (ICA) has been submitted for solving different optimization problems. This algorithm is based on the socio-political competition among empires. This paper proposes a novel ICA algorithm for automated mining of the exciting and readable association rules without considering the minimum support and confidence thresholds. The convergence rate and computational efficiency of ICA have been improved. This study shows that ICA is combined with some operators of genetic algorithms. The experimental results show that this algorithm is more efficient than the methods of mining association rules based on the basic ICA and the genetic algorithm. These modifications are not only valid for association rule mining but also have extensions to other optimization problems.*

Subject Categories and Descriptors: [H.2.8 Database Applications]: Data mining [I.1.2 Algorithms]

General Terms: Data Mining, Algorithms, Genetic Algorithms, Association Rules

Keywords: Association Rules, Imperialist Competitive Algorithm, Genetic Algorithm, Evolutionary Algorithm

Received: 14 March 2020, Revised 28 November 2020, Accepted 13 March 2021

Review Metrics: Review Scale: 0-6; Review Score: 4.32; Inter-reviewer consistency: 81.25%

DOI: 10.6025/jdim/2021/19/4/135-143

1. Introduction

With the development of information technology, there are many different kinds of information databases, such as medical data, financial data, and marketing transaction data. Data mining techniques have been the most widely applied in recent decades to find the critical hidden knowledge from these databases and predict their future behavior. Data mining can be categorized into several tasks, including association rule, sequential patterns, time series, clustering, classification, etc. [1].

Association rules play a significant role in data mining scenarios. They help find out the usage patterns and relationships between items in the database of transactions [2]. A Transactional database refers to the collection of transaction records, usually sales records. When a data set consists of continuous values, it becomes hard to mine the data, and some special techniques need to be prepared [3]. Association rule consists of many modified algorithms based on Agrawal, Apriori, SETM [5], AIS [4], DIC [6], and other methods that focus on improving its efficiency and accuracy.

These algorithms have their limitations, as they have to mine association rules in two stages. The best rules are considered the best rules, whereas most of these rules can easily be predicted by the users. They are not attractive, and the user will never use them. Also, they mine rules with many attributes, which are not understandable for the user. Hence, they lack objectiveness and efficiency.

In recent years, some evolutionary algorithms have been used with multi-objective functions in addition to standard methods. An evolutionary, genetic, ant colony, simulated annealing, particle swarm optimization, and gravitational search algorithm have been used for mining association rules.

To get the objectives mentioned above, the presented work, almost like the method in [7], proposes an efficient imperialist competitive algorithm (ICA) for mining interests and understandable association rules, which does not need the determination of the minimum support and the minimum confidence thresholds. Also, it extracts these rules as a single run, whereas the previous methods mine them in two stages.

This paper emphasizes improving the convergence rate and increasing the exploration of the algorithm to globally optimal solutions. Therefore, some modifications have been made to the basic ICA, such as adding the mutation operator with adaptive rate and improving the modeling of the assimilation policy that the mutation probability is automatically determined in this algorithm; hence this algorithm is very user friendly. In this way, we improve ICA and our proposed method in [7].

The rest of this paper is organized as follows: Section 2 briefly introduces the general preliminaries about the association rules; Section 3 briefly introduces the related works; Section 4 describes the proposed algorithm in detail with an overview on ICA, then the experiments and we will discuss the analyzed results in Section 5; in section 6, there is a final evaluation of MICA. Finally, we will present the conclusion and future works in Section 7.

2. Preliminaries: Definition of Association Rule

While defining the Association Rule, Agrawal et al., in 1993 [4], first proposed an association rule algorithm to analyze the customers' market basket, showing that some hidden relationships exist between purchased items in transactional databases. These results can help make decisions [8]. An association rule is in the form of $A{=>}C$, where A and C represent antecedent and consequent itemsets of the rule, respectively. The general transaction database ($D = \{T_1, T_2, ..., T_n\}$) can represent the possibility that a customer will buy product C after buying product A and $A \cap C = \phi$. $I = \{I_1, I_2, ..., I_n\}$ is a set of all items appearing in the transaction database. Any non-empty subset x from I is called an itemset, an itemset that contains k items is called a k-itemset. The basic framework for mining association rules includes two stages. First, find frequent itemsets; second, generate association rules based on these frequent itemsets.

However, two measures of support and confidence are introduced for evaluating association rules. These are calculated from (1) and (2) equations, respectively [1]:

a) Support ($A{=>}C$): Finding itemsets with their supports above the minimum support threshold which is called "frequent itemset".

$$Sup\ (A{=>}C) = \frac{\text{Number of Transactions which contain } A \text{ \& } C}{\text{Number of Transactions in the database}} \quad (1)$$

b) Confidence ($A{=>}C$): Using frequent itemsets found in (1) to generate association rules that have confidence levels above the minimum confidence threshold.

$$Conf\ (A{=>}C) = \frac{\text{Number of Transactions which contain } A \text{ \& } C}{\text{Number of Transactions which contains } A} \quad (2)$$

These rules which satisfy both minimum support and minimum confidence threshold are called strong association rules [1].

3. Related Works

Association rule mining is one of the essential techniques in data mining. There have been introduced many association rule mining algorithms based on the proposed methods by Agrawal in [4] and [9]. In addition, an approach has been submitted for mining association rules in large relational tables, including both quantitative and categorical attributes [10]. On the other hand, recently optimization methods for mining association rules have been applied.

The idea of using a genetic algorithm has been employed in discovering only frequent itemsets [11]. In another study, a new method using a genetic algorithm has been proposed for finding negative and optimized association rules [3]. In [12], other researchers applied a genetic algorithm for discovering association rules from a manufacturing information system (MIS) dataset. The results of using this method showed that the genetic algorithm is more efficient.

In another study, an ant colony system has been applied for mining multi-dimensional rules. The test results indicated that the proposed method could extract more condensed rules than the *Apriori* method [13]. Later, this method was combined with the clustering method to provide more detailed rules [14]. Also, a method of extracting association rules using a multi-objective genetic algorithm has been proposed [15].

In [16], a method of exploring frequent itemsets has been proposed by combining PSO with the Ant algorithm. In comparison with the GAR algorithm in [11], this method is faster and has more accuracy, but it can discover only frequent itemsets, and GAR has the same limitation. In [8], a method for mining association rules has been proposed using a particle swarm algorithm.

In [17], have been extracted association rules based on the genetic algorithm. In this method, relative confidence was used as the fitness function. But only rules with fixed length were removed. Also, this method's fitness function is so that it is trapped into the local optimum, and hence many rules are generated. Later, other researchers improved it by defining a new multi-objective fitness function [18]. In [19] simulated annealing algorithm has been used to develop a multi-objective rule mining method.

Another study used the gravitational search algorithm to develop a multi-objective association rule mining method [20]. The results of applying this method compared to the proposed method in [8] showed that it is more able to discover global solutions, and individuals evolve into the convergent proper positions with a higher fitness value. In [22, 23, 24, 25, 26], potential mining algorithms are experimentally tested.

There is no research, deploying the imperialist competitive algorithm for mining association rules, except in [7]. Therefore, in this paper, we tried to provide a valuable method for extracting association rules.

4. MICA in Association Rule Mining

This section describes the MICA algorithm proposed in this study for mining interesting, readable and understandable association rules. It is based on one of the latest evolutionary algorithms called the imperialist competitive algorithm (ICA).

The mica is one of the essential materials technology that is compatible with the concept of our algorithm and its application in data mining to explore the exciting and readable of association rules, which is essential in this area; hence, this algorithm has been named MICA (Mining association rules via a novel ICA). Since ICA has been introduced as a method based on the minimum value-finding, in this work, it has been changed into a searcher of the maximum value. The proposed algorithm consists of a series of stages that Figure 1 illustrates the algorithm structure with an overview.

Evolutionary algorithms are not necessarily appropriate for solving optimization problems, and there are different methods for solving an optimization problem by applying a specified evolutionary algorithm.

This paper's main objective is to evaluate the strengths and weaknesses of the ICA algorithm and consider the association rule mining problem, which has been improved. Modified in Figure 1, the margins are shown in bold.

The following sub-sections describe some essential parts of the MICA algorithm: an overview of the imperialist competitive algorithm, country representation, modeling the assimilation policy, and finally, the fitness function of the proposed algorithm is described.

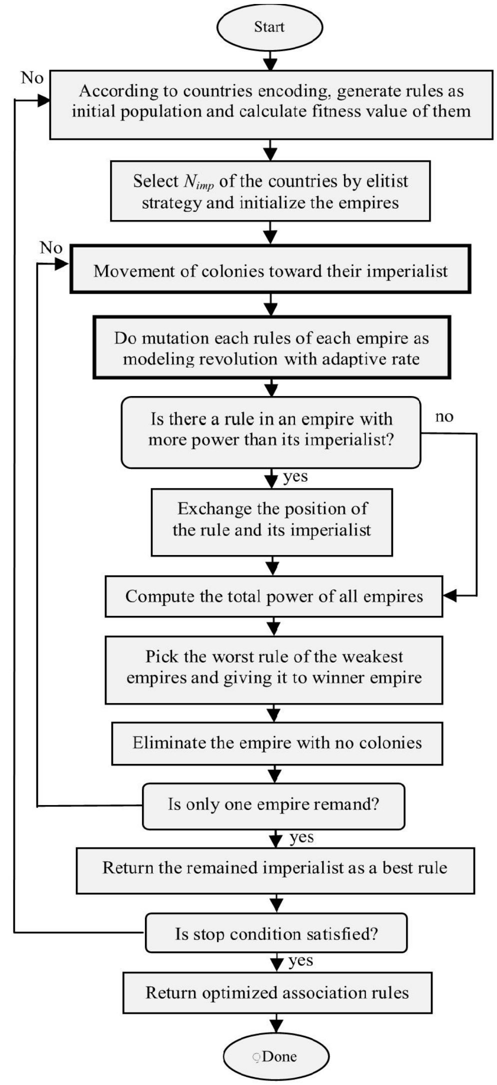

Figure 1. The structure of the proposed algorithm

4.1 An Overview on Imperialist Competitive Algorithm

Evolutionary algorithms are not necessarily appropriate for solving optimization problems, and there are different methods for solving an optimization problem by applying a specified evolutionary algorithm. This paper's main objective is to evaluate the strengths and weaknesses of the ICA algorithm and consider the association rule mining problem, which has been improved. Modified in Figure 1, the margins are shown in bold. The following sub-sections describe some essential parts of the MICA algorithm: an overview of the imperialist competitive algorithm, country representation, modeling the assimilation policy, and finally, the fitness function of the proposed algorithm is described.

Like other evolutionary algorithms, ICA starts with an initial population of individuals, which are called countries. Countries are divided into two groups: imperialists and colonies, which together form empires. The division of all the colonies of initial countries is based upon the fitness value. In this algorithm, the more powerful imperialists, have the more colonies. The colonies in each of the empires started moving towards their imperialist countries based upon a simple model of assimilation policy. The total power of an empire is defined by the power of the imperialist country and the percentage of mean power of its colonies. Then the imperialistic competition among these empires forms the proposed evolutionary algorithm. During this competition, an empire that cannot succeed in this competition cannot increase its power shall be eliminated. Based upon this competition, the power gradually increases for some empires and decreases for others. This results in the collapse of weak empires. Finally, the movement of the colonies toward their relevant imperialist states along with competition among empires and also the collapse mechanism caused all the countries to converge to a condition in which their existence just one empire in the world and all the other countries are colonies of that empire. Imperialistic competition converges to a state where only one empire and colonies have the same cost function value as the imperialist [21].

4.2 Countries Representation

In this study, the countries being produced and modified along the evolution processes represent association rules. Each country consists of dimensions that represent the items in the dataset. A positional representation, where the j^{th} item is encoded in the j^{th} dimension, has been used. According to Figure 2 in the proposed algorithm, every country has n decision variables in lieu of n items in any dataset. n items in any dataset.

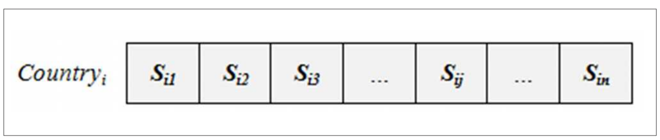

Figure 2. Countries representation

Each dimension has one part that represents the antecedent or consequent of the item in the association rule and can take three values: ‘0’, ‘1’ or ‘2’. If the dimension is ‘0’, this item will be in the antecedent of the association rule, and if it is ‘1’, this item will be in consequence of the association rule. If it is ‘2’, this item will not be involved in the association rule.

4.3 Modeling the Assimilation Policy

Moving colonies toward the relevant imperialist is the motivation of the assimilation policy. This process means that one of the weaker individuals can move toward a better individual. It causes the colonies to be like the relevant imperialist. In this work, for modeling the assimilation policy in the mining association rule problem, a method has been introduced that is almost like the proposed method in [7].

In this way, some attributes of the relevant imperialist are copied randomly in the colony for each colony. But unlike the proposed method in [7], here, these attributes are randomly selected from among different bits between the colony’s rule and the imperialist rule, which leads to the increase of the convergence of the algorithm and the reduction of the re-computation time. Therefore, this approach is more appropriate.

In this method, first, the amount of d as the distance between the colony’s rule and the imperialist rule is calculated for each colony. *d* is the number of different bits between the colony’s rule and the imperialist rule. For example, in Figure 3, other bits between the Imperialists and the colony have been determined; it also shows that this distance between them is four.

Then, the random number n, between 0 and *d* with uniform distribution, is selected for each colony. Then, *n* bits. Of the colony are changed randomly, which are different between the colony’s rule and the imperialist rule. It means that all bits selected as ‘0’ convert to ‘1’; all chosen bits as ‘1’ convert to ‘2,’ and all chosen bits as ‘2’ convert to ‘0’.

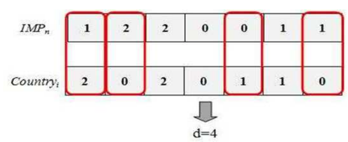

Figure 3. Calculate the distance between the imperial’rule and the colony’ rule

4.4 Modeling the Revolution with Adaptive Rate

In contrast to the assimilation policy, the imperialists seriously followed it, and the events didn’t happen according to their policy. Here, some colonies of an empire have deviated.

In this work, ICA has been combined with a mutation operator of a genetic algorithm for modeling the revolution. Each colony of an empire is selected randomly, with the chance of a mutation occurring if the number is lower than the mutation rate. This affects the support, confidence, and fitness value of that rule. This method moves the position of empires to the globally optimized solution. Unlike the other genetic algorithms and the proposed method in [7], this method does not require specifying the mutation rate before the start of the algorithm, and its value is calculated for each colony automatically as (3):

$$P^{n}_{m_{col_i}} = 1 - \left| \frac{fit^{n}_{col_i}}{\sum_{j=1}^{N_{col} \in n} fit^{n}_{col_j}} \right| \qquad (3)$$

fit^{n}_{coli} is the fitness of i^{th} colony. P^{i}_{coli} is the mutation probability for i^{th} colony. $1 - P^{i}_{coli}$ is the normalized cost of the *i*th colony. Since the weaker colonies are more likely to be collapsing, in this method, the deviant behaviour of these colonies will be automatically more. This is done to move the position of weak colonies from the negative spaces of the problem toward better solutions and finally increase the power of the empire.

In all algorithm procedures, after any changes in the dimensions of an individual in the population, its rule validity will be measured. That rule with at least one attribute in the antecedent of the rule and one in consequence of the rule is valid..

4.5 Fitness Function

The fitness function should be determined to the specific search spaces; thus, the choice of the fitness function is critical to get the desired results. This paper focuses on mining interesting and readable rules.

4.5.1 Interestingness

Mining association rule is a task that extracts some confidential information, it must discover those rules that have a comparatively less occurrence in the entire database and learning such rules are more complex. For classification rules, measures like information gain can be helpful, but it is not efficient for evaluating the association rules. Therefore, interesting measure in [15] is used in the fitness function which is computed by (4)

$$Interestingness_i = [\frac{\sup_i(A \cup C)}{\sup_i(A)}] \times [\frac{\sup_i(A \cup C)}{\sup_i(C)}] \times [1 - \frac{\sup_i(A \cup C)}{|R|}] \quad (4)$$

This relation has three parts: $[sup_i(A \cup C)/sup_i(A)]$ indicates the probability of creating the rule ith depending on the antecedent part; $[sup_i(A \cup C)/sup_i(C)]$ shows the probability of creating rule i^{th} depending on the consequent part. Most of these are exciting 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 is used.

In the third part of this parameter, $[sup_i(A \cup C)/|R|]$ gives the probability of generating the rule i^{th} depending on the whole dataset. In this parameter $|R|$ is the total number of records in the database. So complement of this probability will be the probability of not generating the rule. Because, those rules have a very high support count and high frequency will be less attractive, and such rules are easily predictable by the users.

4.5.2 Readability

Readability is another objective used in MICA. This parameter rewards the shorter rules with a smaller number of attributes. Readability and comprehensibility of rules that are important in data mining are increased. It is known that larger rules have more redundant information.

In result, it is difficult to understand the rules and the user will never use them. Readability has been computed by (5) where $|attributei(A \cup C)|$ is the number of attributes that exist in total rule i and |*attribute*| is the total number of attributes in database.

$$readability_i = 1 - \frac{|attributes_i(A \cup C)|}{|attributes(D)|} \quad (5)$$

In this paper, the weighted sum fitness function is calculated for every population's individual by (6). All objectives have been weighted in order to give them different importance.

$$fitness_i = \sum \omega_j O_{ij} \ , \quad 0 \le \omega_j \le 1 \quad (6)$$

Here, ω_j is the weight for objective j, and O_{ij} is the fitness of the j^{th} objective of rule i. Each w_j will be specified by the percentage of the user's interests, and one might increase or decrease the effects of parameters of the fitness function. It means that they do not need to assign for each database in spite of some thresholds such as minimum support and minimum confidence in the previous methods. Also, it should be noted that another objective can be added.

5. Implementation and Experimental Results

MICA algorithm was implemented and executed in C#, on a PC with Intel Dual-Core 2.1 GHz operator on a 2GB Ram. The setting of the used parameters is shown in Table 1. The values of the weighted coefficients for the interestingness and readability objectives which have been introduced in the proposed fitness function were selected as 0.9 and 0.1, respectively. These coefficients are specified by the percentage of user's interests. Hence, they do not need to re-initialize for each database in spite of some thresholds such as minimum support and minimum confidence.

$N_{Country}$	N_{Gen}	N_{Imp}
70	100	10

Table 1. The used parameters values for running MICA

5.1 Real Databases

The proposed method was executed and evaluated on four real datasets: balance scale, nursery, car evaluation, basket market. For doing examinations we used the basket market dataset existing in Clementine 12.0 tool [22] and three datasets in UCI at http://www.ics.uci.edu/

Dataset	$N_{Records}$	$N_{Attributes}$
Balance Scale	625	23
Nursery	12960	32
Car Evaluation	1728	28
Basket Market	1000	11

Table 2. The specification of datasets

~mlearn. The specifications of datasets used are given in Table 2.

All attributes in datasets are categorical except basket market dataset; therefore we converted them into the Boolean datasets. It means that every attribute with any amount is considered as an item. The basket market dataset includes 11 binary attributes and it has the least attributes in this work. Nursery dataset has the most records over than 10000 examples and also, it has the more attributes than the rest of the dataset.

5.2 Experiments and Analyzing the Results

The first experiment is evaluating the convergence rate and the performance of the proposed algorithm to discover the best rules by considering the introduced objectives in this study. Therefore, MICA is compared with MINICA [7] and the method of mining association rules which is based on the GA [15]. The initial setting of the used parameters values for the GA [15] is given in Table 3.

N_{POP}	N_{Gen}	$P_{Crossover}$	$P_{Mutation}$ (5 point)
70	100	0.8	0.02

Table 3. The used parameters values for the proposed GA

In order to compare the experiments fairly, we examined them by the presented fitness function in this work. Also, we examined MINICA algorithm with the same parameters as in Table 1. This experiment was done with 20 replications. So, the mean of the results was calculated. Figures 4 to 7 illustrate the convergence speed and accuracy of these algorithms and the lines in these figures show the fitness of the best imperialist in each generation by any of the three algorithms.

As it is depicted in Figure 4, MINICA and MICA algorithms have a better performance than the GA algorithm. Although, MINICA has a better fitness at the 15th iteration, finally, MICA has a better ability to discover the best rules.

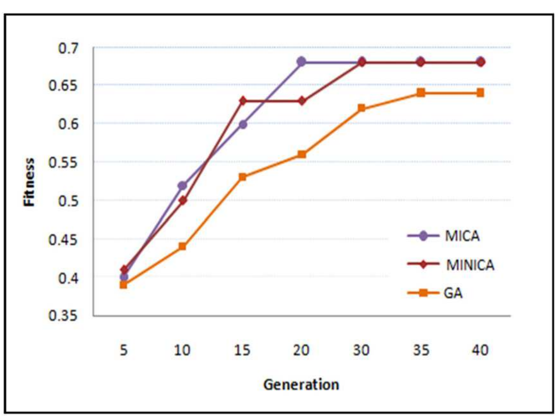

Figure 4. Convergence rate on the Balance Scale dataset

In the Figure 5, the performance of MINICA on the car evaluation dataset is better than the GA [15] but MICA indicates a better performance both in convergence speed and in obtaining the quality of the association rules in compaction to the other two algorithms.

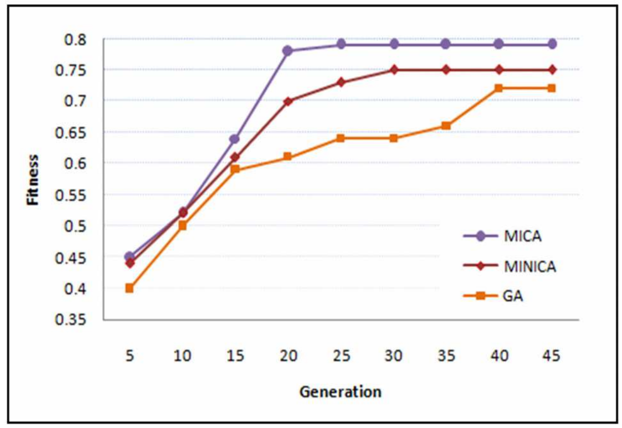

Figure 5. Convergence rate on the Nursery dataset

According to Figure 6, at the first 10 iterations, the GA algorithm has a better fitness than MICA and MINICA algorithms, but from the 15th iteration to the end, two other algorithms have been achieved to better rules. As it can be seen, MINICA has converged faster than MICA. But it is not an important issue, because MICA has been able to escape from the local solutions and reach to global optimal solutions.

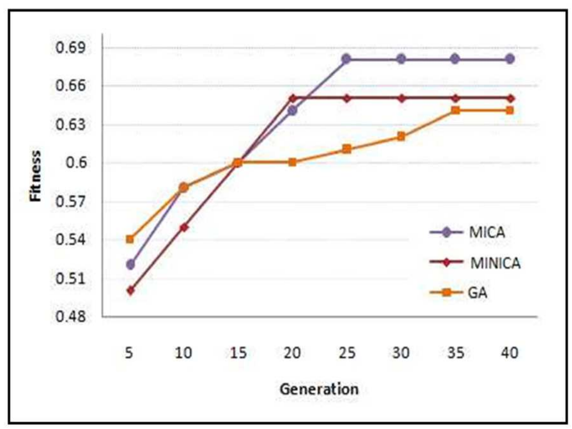

Figure 6. Convergence rate on the Car Evaluation dataset

In Figure 7, it is observed that MICA and MINICA have a better performance in the optimal solution quality and convergence speed than the GA algorithm. Also, MICA has a better convergence speed in comparison to MINICA. These figures depicted that the modifications applied are beneficial.

The results of the final test in this paper have been shown in Table 4, which is about the comparison of the details of the mean results obtained from MICA, MINICA [7] and Figure 7. Convergence rate on the Basket Market dataset the proposed GA in [15]. The mean number of best

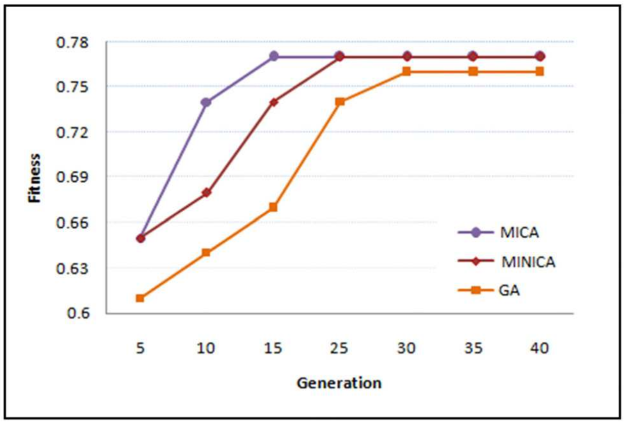

Figure 7. Convergence rate on the Basket Market dataset

different mined rules, the mean number of attributes contained in the rules, the mean of the readability, support, confidence, and the cosine value of these rules are shown in this table.

The cosine measure can be used for evaluating the interestingness of the association rules. According to this table, MICA presents better results in comparison with MINICA; on the other hand, MINICA has better results in comparison with the association rules mining method in [15]. MICA and MINICA algorithms could discover rules with appropriate support and a high confidence value compared to the GA in [15]. Also, the number of attributes obtained in the rules by these two methods is smaller; hence, the readability of these rules is more. As mentioned before, more extensive rules are more likely to contain redundant information.

In addition, in these two methods, the mean cosine value which can be used for evaluating the interestingness of the rules is more than the GA [15]. On the other hand, with a closer look at the obtained mean results by MICA and MINICA, we can say that MICA can provide better results on some datasets (such as nursery and car evaluation that have prominent attributes and records) with a more difference. It can give the desired results on other datasets with a bit of difference.

6. Evaluation of MICA

As seen, MICA and MINICA algorithms based on ICA have better performance with the slightest error in comparison with the algorithm based on the GA for mining association rule problems.

In ICA, individuals have more interacted with each other, and thus, these algorithms are more likely to discover a global solution. Also, MICA has better efficiency than MINICA algorithm both in the quality of optimal solutions and in convergence speed. On the other hand, it has better ability to avoid hasty convergence because MICA is based on the improved ICA, which has been introduced

Parameters	Datasets				
	Methods	Balance Scale	Nursery	Car Evaluation	Basket Market
Support (%)	MICA MINICA GA [15]	7.22 7.24 5.32	18.11 18.21 17.55	6.12 6.14 5.61	15.88 15.87 14.38
Confidence (%)	MICA MINICA GA [15]	57.10 57.02 48.37	71.04 69.02 67.42	45.68 43.54 39.33	58.27 58.24 52.77
$N_{Attributes}$	MICA MINICA GA [15]	2.5 2.5 4.2	9.1 9.3 9.4	3.1 3.2 3.9	2.3 2.3 3.2
Readability	MICA MINICA GA [15]	0.89 0.89 0.80	0.71 0.71 0.70	0.90 0.89 0.86	0.79 0.79 0.70
N_{Rules}	MICA MINICA GA [15]	31 32 28	35 35 35	20 19 22	10 10 9
Cosine	MICA MINICA GA [15]	0.48 0.47 0.39	0.59 0.56 0.52	0.44 0.38 0.33	0.59 0.57 0.51

Table 4. Comparisons of the results

In this work by the generalization of the basic ICA. In fact, by modification of the modeling of the assimilation policy and by combining ICA with a mutation operator of a genetic algorithm with adaptive rate, an individual with a higher fitness searches the space with slow movements and hence explores the space more locally. Also, an individual with a lower fitness searches the space with rapid movements and searches the space more globally.

7. Conclusion and Future Works

This paper proposed a novel ICA algorithm for mining association rules without considering the minimum support and the minimum confidence thresholds, which are difficult to determine for each database. The algorithm extracts automatically the readable and engaging rules as a single run. In addition, other objectives can be added to the fitness function. This algorithm has been called MICA, which is extending MINICA algorithm.

This paper emphasized more on the improvement of the performance both in convergence rate and in obtaining the quality of the association rules, so to achieve these purposes, several modifications have been presented that includes modifying the modeling of the assimilation policy and combining that algorithm with a mutation operator of a genetic algorithm with adaptive rate. In this way, modeling the assimilation policy and modeling the revolution in ICA and our proposed method in [7] have been improved. MICA's performance was evaluated through several tests in various real datasets. The results of all experiments in this work indicate that MICA has a better ability to discover global solutions with a fast convergence rate compared to MINICA and the mining association rules based on the GA [15].

Regarding that MICA algorithm has an appropriate structure for parallel architecture, the parallelization of this method can be helpful to reduce the run-time. Moreover, in this paper, for association rule mining, ICA has been improved; this improved version can be applied to other optimization problems as future work. In addition, this algorithm can be extended for mining the weighted association rules.

.

References

[1] Han, J., Kamber, M. (2006). Data Mining: Concepts and Techniques*,* Morgan Kaufmann in Elsevier, Second Edition, San Francisco.

[2] Agrawal, R., Imielinksi, T., Swami, A. (1993). Database mining: a performance perspective*, In*: *Proceedings of IEEE Transactions on Knowledge and Data*, 5 (6) 914–925.

[3] Saggar, M., Agrawal, A.K., Lad, A. (2004). Optimization of association rule mining using improved genetic algorithms*, In.*: Proceedings of the IEEE International Conference on Systems Man and Cybernetics, p. 3725–3729.

[4] Agrawal, R., Imielinski, T., Swami, A.N. (1993). Mining association rules between sets of items in large databases, *In.*: Proceedings of ACM SIGMOD international conference on Management of data, Washington, 22 (2) 207–216.

[5] Houtsma, A., Swami, M. (1993). Set-oriented mining of association rules, Research Report RJ 9567, *IBM Almaden Research Center*, San Jose, California.

[6] Birn, S., Motvani, R., D.Ullman, J., Tusr, S. (1997). Dynamic itemset counting and implication rules for market basket data*, In*: Proceedings of the ACM SIGMOD international conference on Management of data, New York, 26 (2) 255–264.

[7] Khademolghorani, F. (2011). An effective algorithm for mining association rules based on Imperialist Competitive Algorithm*, In*: Proceedings of the sixth International Conference on Digital Information Management (ICDIM 2011), Melbourne, Austria, p. 6-11.

[8] Kuo, R.J., Chao, C.M., Chiu, Y.T. (2009). Application of particle swarm optimization to association rule mining*,* Elsevier Applied Soft Computing, 11(1) 326-336.

[9] Agrawal, R., Srikant, R. (1994). Fast algorithm for mining association rules in large databases*,* Research Report RJ 9839, IBM Almaden Research Center, San Francisco, CA, USA, p. 487-499.

[10] Srikant, R., Agrawal, R. (1996). Mining quantitative association rules in large relational tables*, In.*: Proc. of ACM SIGMOD international conference on Management of data, Montreal, 25 (2) 1-12.

[11] Mata, J., Alvarez, J., Riquelme, J. (2002). Discovering numeric association rules via evolutionary algorithm*, In.*: Proc of sixth Pacific–Asia conference on knowledge discovery and data mining PAKDD-02 (LNAI), Taiwan. p. 40-51.

[12] Li, C., Yang, M. (2004). Association rule data mining in manufacturing information system based on genetic algorithms*, In*: Proceedings of the 3rd International Conference on Computational Electromagnetics and Its Applications, p. 153–156.

[13] Kuo, R.J., Shih, C.W. (2007). Association rule mining through the ant colony system for National Health Insurance Research Database in Taiwan*,* Computers an Mathematics with Applications, Tarrytown, NY, USA, 54 (11-12) 1303–1318.

[14] Kuo, R.J., Lin, S.Y., Shih, C.W. (2007). Discovering association rules through ant colony system for medical database in Taiwan*, International Journal of Expert Systems with Applications*, 33 (3) 794-808.

[15] Ghosh, A., Nath, B. (2004). Multi-Objective Associa tion Rule Mining Using Genetic Algorithm*,* Elsevier Information Sciences, 163 (1) 123–133.

[16] Badawy, O.M., Habib, M.I., Sallam, A.A. (2008). Quantitative Association Rule Mining Using a Hybrid PSO/ACO Algorithm (PSO/ACO-AR)*, In*: Proceedings of Arab

Conference on Information Technology (ACIT'2008), Hammamet, Tunisia, p. 1-9.

[17] Yan, X., Zhang, Ch., Zhang, Sh. (2009). Genetic algorithm-based strategy for identifying association rules without specifying actual minimum support, processing of Expert Systems with Applications, 36 (2) 3066-3076 .

[18] Qodmanan, H., Nasiri, M., Minaei-Bidgoli, B. (2010). Multi objective association rule mining with genetic algorithm without specifying minimum support and minimum confidence*, In.*: Proceedings of Elsevier Expert Systems with Applications, 38 (1) 288-298.

[19] Nasiri, M., Taghavi, L., Minaei-Bidgoli, B. (2010). Multi-Objective Rule Mining using Simulated Annealing Algorithm*,* Soft Computing in Journal of Convergence Information Technology (JCDA), Korea, 5 (1) 60-68.

[20] Khademolghorani, F., Baraani, A., Zamanifar, K. (2011). Efficient mining of association rules based on gravitational search algorithm*,* IJCSI International Journal of Computer Science Issues, Mahebourg, 8 (4) 1-8.

[21] Atashpaz-Gargari, E., Lucas, C. (2007). Imperialist Competitive Algorithm An Algorithm for Optimization Inspired by Imperialistic Competition*, IEEE Cogress on Evolutionary Compution*, Singapore, p. 4661-4667.

[22] Rose Tinabo. (2011). A Mechanism for Selecting Appropriate Data Mining Techniques, *Journal of Intelligent Computing*, 2 (1) 35-41. March.

[23] Quanyin Zhu., Pei Zhou., Suqun Cao., Yunyang Yan., Jin Ding. (2012). A Novel RDB-SW Approach for Commodities Price Dynamic Trend Analysis Based on Web Mining, *Journal of Digital Information Management*, 10 (4) 168-173. August.

[24] Yang Hang., Simon Fong. (2011). Algorithmic level stream mining for Business Intelligence System Architecture building, *International Journal of Web Applications*, 3 (1) 29-35. March, 2011.

[25] Mohamed El Ghourabi., Amira Dridi., Fedya Telmoudi. (2011). Data Mining versus Statistical Tools for Value at Risk Estimation, *Journal of Information Technology Review*, 2 (4) 154-162. November, 2011.

[26] Hassan I. Abdalla. (2011). New Technique to Deal with Dynamic Data Mining in the Database, *Journal of Digital Information Management*, 9 (4) 147-152. August 2011.

Author biographies

Fariba Khademolghorani obtained her BS degree in Computer Engineering in 2007 and her MS degree in 2011 at the Faculty of Computer Engineering of Islamic Azad University of Najafabad in Iran. Her research interests include: swarm Intelligence, data mining, soft computing and evolutionary algorithms. She published over eight papers in data mining in national journals and national conferences. Also, she proposed two papers in interna tional journal and international conference. She continues her researches in the field of association rules mining from databases with different types of attributes by using evolutionary algorithms.