International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 10, No.1 (Feb-2021)
http://dx.doi.org/10.12785/ijcds/100133

Whale Optimization Algorithm for Solving Association Rule Mining Issue

KamelEddine Heraguemi[1], Houria Kadrii[1] and Amira Zabi[1]

[1] *Department of computer science, University of M'sila, Algeria*

Received 14 Jul. 2020, Revised 3 Aug. 2020, Accepted 16 Aug. 2020, Published 8 Feb. 2021

Abstract: Our-days, with the significant number of connected devices, data stored has grown significantly. The exploitation of the information stored in it for decision-making has become indispensable in the most prominent companies. Association rule mining draws in consideration of researchers to extract relationships between items in data. Nevertheless, this process is costly in terms of memory and computation, Which are the foremost drawbacks in exact algorithms designed for mining association rule. Recently, swarm-based algorithms prove their efficiency in data mining. With this in mind, an updated whale optimization algorithm to mine association rules, called (WO-ARM), proposed in this paper. Whale optimization has a good trade-off between intensification and diversification, which inherited by our proposal. A set of tests carried out on different famous benchmarks. The preliminary results show that our approach beats other swarm inspired algorithms already exist in the literature in terms of quality, runtime, and memory usage.

Keywords: Association rule mining, Whale optimization algorithm, Support,, Confidence, swarm inspired algorithm

1. INTRODUCTION

Nowadays, stored data have grown due to the enormous development of the INTERNET and unlimited connected devices. All these stored data are used by several users and companies to generate useful information to help for decision making. Therefore, data processing turned into a real challenging issue, which imposes the development of new frameworks and methods with low processing time and low memory usage. The most frequently used method to process data in the last decade is Knowledge Discovery in Databases (KDD), which aims to extract interesting patterns from stored data, commonly contains three successive stages: Pre-processing; Data Mining; and finally, Post-Processing. Within KDD, the primary process is data mining that has a goal to extract non-trivial information hidden in data. It contains several techniques used commonly in data processing such as Classification, Clustering, Regression analysis, and Association rules [1].

Association rule (AR) has attracted researchers' attention since its first release by Agrawal et al. in the early 90s [2]. It refers to relationships that exist between items in a real-world database. It was designed initially for market basket analysis to obtain relations between products, like *milk* ⇒ *bread*, which means that someone bought milk, also get bread with high probability. These rules would allow managers to plan their marketing strategy to increase benefits. In the last decade, ARs become very utilized in different application domains such as medical diagnosis, biomedical literature, protein sequences, logistic regression, and fraud detection on the web, etc. Mining association rules is the process that generates relationships among items in a data-set that generally given as If-THEN statements. Restrictions are in If statement, and those inside THEN clause are Outcomes. Many traditional algorithms have been developed to solve ARM issues, such as Apriori[3], FP-growth [4], Etc. These algorithms created to extract all the relations that exist in the dataset. However, they suffer, our days, from the considerable quantity of data stored in databases that affect their execution time and memory usage. In order to overcome exact algorithms drawbacks, researchers apply intelligent meta-heuristic, which are previously employed to solve numerous NP-Complete problems, In which ARM problems can be classed. As an NP-complete problem, many works proposed to use evolutionary algorithms and swarm-inspired algorithms to solve Arm to pick the optimal rules. Firstly, genetic algorithm[5] has been successfully applied and given promising results.. Few years after swarm intelligence was employed with ARM using various well-known algorithms such as

E-mail: kameleddine.heraguemi@univ-msila.dz, houria.kadri@univ-msila.dz, Amira.zabi@univ-msila.dz

Particle Swarm Optimization [6], Bees swarm Optimization (BSO) [7], Bat algorithm (BA) [8]..Etc. Formally, The datasets regarded as sample search space, in which the algorithm tries to maximize/minimize an objective mathematical function that compute the selected rules quality according to several measurements.

The whale optimization algorithm is newly swarm-based algorithms produced by Mirjalili et al.[9]. It simulates the whales' humpback hunting behavior. Usually, humpback whales hunt fishes near sea surface by moving around the victim and produce bubbles circle or 9-formed path. This method is a specific hunting technique for humpback whales called the bubble-network feeding method. Many pieces of research have been conducted on WOA in the last three years. Those works applied WOA in various real-world optimization problems utilizing many ways, such as improvements, hybridization, and proposing new variants for the algorithm[10]. WOA confirmed its competitivity in-the-face of other swarm inspired metaheuristics such as PSO, BA, and BSO in terms of exploitation by exploration. Which make WOA one of the most utilized in numerous domain as: Electrical Engineering[11], Classification[12], Clustering[13], Image Processing[14], and many other problems. Highly motivated by the success of WOA, this paper suggests a new whale optimization algorithm to deal with the association rule mining problem named WO-ARM, to extract high-quality rule that can be useful for the final user. This proposal investigates the advantages of the whale algorithm. Firstly, with its simplicity and low complexity, it will utilize lower computing power and less memory. Also, WOA needs a low number of parameters that make it suitable to use for final users. In order to judge the stated WO-ARM method, profound experimental tests are carried out on various datasets benchmarks with different sizes. Our initial outcome are encouraging. Also, the proposed algorithm demonstrates its effectiveness compared to similar methods in the Association Rule Mining field according to runtime consumption and rules quality.

The remainder of this article designed as follows: in the next piece of writing will resume the state-of-art regarding novel work in the association rule mining field and the different evolutionary algorithm applied to solve this mentioned problem. In the third section, a general background about the rule mining problem and its principals presented. Furthermore, the section will introduce the original whale optimization algorithm. After, Section 4 outlined our proposal. Our experimentation and results interpreted in the fifth part. Eventually, we will draw some conclusions.

2. RELATED WORK

Association rule mining problem has been largely studied since it first appears in 1993 by Agrawal and co-workers [2]. A surprising number of studies can be found in the literary study, that can be separated into two principal classes: exact and optimization methods. The first class aims to extract all the relationships between items exists in all the database, whereas the other has a goal to generate the primary and useful rules to the final user.

Many exact algorithms extract association rules from various types of databases. Three significant methods have dominated associated rules mining: The popular conventional algorithm named apriori discovers the whole relationships based on minimum support defined by the final user. [3], FP-growth developed to solve with Apriori drawbacks, mainly the dataset scan many times, where it is just the whole database just two times [4]. Afterward, various works investigated, in apriori and Fp-growth difficulties, have viewed the light such as Eclat [15],Charm[16]. This class of algorithms suffers our-days with a large amount of data that making them slower and memory consumers. Hence, researches go over the second class, which is recently developed. In the last ten years, more tens of papers are published with new optimization methods deal with the association rule mining problem.

The first investigation with association rule mining as an optimization problem, the genetic algorithms used [5]. The authors in [17] proposed a tool called GENAR (GENetic Association Rules) that discovered relationships in a quantitative database based on a genetic algorithm (evolutionary algorithm). As first work, results improve the utility of an evolutionary algorithm in such a problem, which opens the door to various other works. Haldulakar et al. applied the genetic algorithm to optimize apriori algorithm results to generate the most reliable and most helpful rules for the final user.[18]. Another work applied a genetic algorithm to resolve the ARM issue without specified minimum support and minimum confidence called ARMGA[19]. This approach returns numerous invalid chromosomes (rules) and produces a sizable amount of rules. In [20], the authors utilized the extracted association rules by a genetic algorithm to discover a powerful association amongst several leading factors. The outputs give significant rules in less time without specifying any support or confidence measures. Most lately, the produced associations rule based on a genetic algorithm, are used to improve collaborative filtering recommendation system[21]. That is one of the most critical applications of association rules. Another application of ARM is presented in [22], in which the authors applied a genetic algorithm to extract rules from numerical data and use the results for Smart Cities. Genetic algorithms prove their efficiency in many applications, including the association rule mining problem. Nevertheless, the critical obstacle with it is the essential parameters selection, such as mutation and crossover rate, and the selection criteria of the new chromosomes should carry out correctly.

In the last few years, with the outgrowth of bio-inspired algorithms. Many swarm intelligence algorithms

as PSO algorithm, Bat algorithm, Firefly algorithm. Etc, are suggested to deal with association rule mining problem. In [23], PSO used to find association rules. In which two stages took place: preprocessing step and mining step. The first portion measures the fitness, whilst, in other, PSO is employed for rules extraction. An improved work based on PSO is developed in [24]. This work provides a binary version of PSO used to mine association rule called (BPSO), which the best X rules generated without undertaking any measures thresholds, whereabouts *X* is a performance threshold. Few other works in the literature based on Bee swarm optimization algorithms, in which the authors presented a method, called BSO-ARM,[25] for Association Rule Mining. The upshots proved that BSO-ARM is more trustworthy than evolutionary algorithms previously developed. Moreover, an enlargement of this work published with three procedures to discover the exploration area of every bee (modulo, next, syntactic). These change of state lifted the quality of rules extracted[7]. In [26], The writers proposed an Association Rule Miner based on penguins search optimization algorithm (Pe-ARM). This method differentiated by a good exploration over the search space. The authors tested their approach to biological data-sets, which proves its efficiency.

Bat Algorithm (BA)[27] is a well-known swarm intelligent algorithm developed by Yang in 2010 to solve continuous optimization problems. An updated version of the bat algorithm proposed in [8] called BAT-ARM, in which the authors defined a new bats' movement formulation according to Association Rule Mining issue fundamentals. The final result proved the efficiency of BAT-ARM, but it stays suffer from the lack of communications between bats that reduce the exploration of the algorithm. To overcome this drawback, the same authors divide the population into different sub-populations and present a master/slave plan to improve BAT-ARM[28]. The outcome surpass those of BAT-ARM in both runtime and rules quality . Later, in [29], two other communication plans introduced for the multiswarm bat algorithm for ARM called ring and Hybrid. More recently, a binary cuckoo search rule miner is presented[30]. In this proposal, the authors modified the cuckoo search algorithm to a binary algorithm based on the sigmoid function and applied it to the ARM issue. A fresh survey outlined the whole domain based heuristic methods is in [31].

Due to the numerous evaluation measurements for association rules, many pieces of research handled association rules as a multi-objective optimization issue by defining different objective functions to maximize measurements. In [32], the authors proposed three multi-objective methods to deal with rule mining by optimizing various quality measurement. Additionally, the paper [33] presented a multi-objective evolutionary algorithm to generate a small number of new rules based on several judgment measurements.. As a complement to their work on bat-inspired algorithm for association rules, Heraguemi et al. [34] suggested a multiobjective BA to extract rules as a four-objective issue using Pareto front solutions. The objective measurements considered to be maximized (interestingness, comprehensibility support, and confidence). Recently, the authors in[35] put forward to extract rules by combining a multidimensional and multiobjective double assembly discrete Firefly Algorithm (MODGDFA) with Pareto rules. The results showed that the generated rules by multi-objective optimization methods are more appropriate than single-objective ones, especially in rules quality, whereas they are slower than single-objective approaches.

3. BACKGROUND

A. Association rule mining

In 1993, agrawal et al. introduced association rule mining problem[36], to extract typical business decisions for helping supermarket managers to design coupons, place products on shelves in order to maximize the profits. These decisions are taken based on the relationships generated from a large amount of transaction history collected over time from sells.

Formally, association rule problem is defined as follow: Let $I = \{i_1 , i_2 , ..., i_n \}$ be a set of literals called items, let $D\{t_1 , t_2 , ..., t_m \}$ be a transactional database where each transaction t contains a set of items. An association rule is implication like $X \Rightarrow Y$ where $X, Y \in I$ and $X \cap Y = \emptyset$. The item-sets X, Y are named antecedent and consequent, respectively. In order to evaluate the generated association rules from any data-set. Also, to detect the most impressive states to the final user, many measurements are invented in the literature, divided into two main groups: objectively and subjectively[37]. The first one involves statistical analysis of the data, whereas the others more oriented towards the user requirements.

In this work, the rule measurements used to evaluate the generated rules. Due to the vast number of patterns extracted from a size-able transnational database, a detected rule is accepted as association rule if its support and confidence are equal or superior to the minimum threshold, specified by a final user, and rejected otherwise. Support and confidence are two measures that aim to determine rules quality which is defined as follows:

Definition 1 "Support is the proportion of transactions in D that contains X, to the total of records in database. Support of item X is calculated using equation (1) and The support of an association rule $X \rightarrow Y$ is the support of $X \cup Y$" [29].

$$support(X) = \frac{Numberoftransactionscontaining X}{TotalNumberoftransactions} \quad (1)$$

Definition 2 "Confidence is the proportion of transactions covering *X* and *Y*, to the total of records containing *X*. When the percentage exceeds a threshold of confidence, an interesting association rule can be generated " [29]. An association rule $X \rightarrow Y$ with a confidence of 80 % means that 80 % of the transactions that contain X also contain Y . The rule confidence is calculated as follows:

$$confidence(X \Rightarrow Y) = \frac{support(X \cup Y)}{support(X)} \qquad (2)$$

B. whale optimization algorithm

In 2016, Mirjalili et al. proposed a new swarm-based nature meta-heuristic inspired by hunting behavior of whales humpback, which considered as the biggest mammals in the world, termed Whale Optimization Algorithm [9]. More precisely, the algorithm mimics the bubbles-net feeding in the foraging behavior of the humpback whales. The bubbles-net formed when the whale swims in a 9-shipped path. Fig.1 shows the Bubble-net feeding behavior.

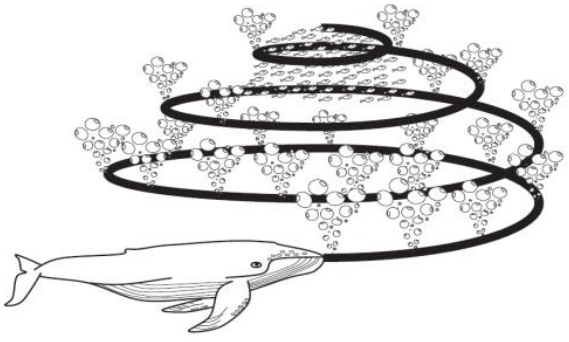

Figure. 1. Bubble-net behavior of humpback whales

As described in Mirjalili et al. paper [9], the algorithm has mainly three phases, Encircling prey, Bubble-net attacking method and Search for prey. These three phases award a good trade-off between exploitation and exploration in the algorithm. Therefore, WOA proves its efficiency in face of other swarm inspired meta-heuristics. The mathematical model of WOA is described as follows:

To hunt a prey, humpback whales first encircle it. Eqs. 3 and 4 can be used to mathematically model this behavior .

$$\vec{D} = \left|\vec{C}.\overrightarrow{X'}(t) - \vec{X}(t)\right| \qquad (3)$$

$$\vec{X}(t+1) = \overrightarrow{X'}(t) - \overrightarrow{(A)}.\vec{D} \qquad (4)$$

where *t* indicates the current iteration, *X'* represents the best solution obtained so far, *X* is the position vector, In addition, *A* and *C* are coefficient vectors that are calculated as in Eqs. 5 and 6.

$$\vec{A} = 2\vec{a}.\vec{r} - \vec{a} \qquad (5)$$

$$\vec{C} = 2.\vec{r} \qquad (6)$$

where $\vec{a}$ decreases linearly from 2 to 0 over the course of iterations (in both exploration and exploitation phases) and $\vec{r}$ is a random vector generated with uniform distribution in the interval of [0,1]. Search agents update their positions based on the best known solution. The solution location is controlled by the adjustments of $\vec{a}$and $\vec{r}$ values.

The hum-pack hunting method is based on shrinking encircling mechanism and a spiral-shaped path toward the prey. The shrinking behavior is formulated as shown in Eq 7.

$$a = 2 - t\frac{2}{\text{MaxIter}} \qquad (7)$$

where t is the iteration number and MaxIter is the maximum number of allowed iterations. The spiral-shaped path is calculated by the distance between the actual solution and the best position by Eq 8,

$$\vec{X}(t+1) = D'e^{bl}.\cos(2\Pi l) + \overrightarrow{X'}(t) \qquad (8)$$

Where $D' = \left|\overrightarrow{X'}(t) - \vec{X}(t)\right|$ describe the distance of i^{th} whale from the prey(The best solution obtained so far). A random coefficient *p* between 0 and 1 is used to choose between the two mechanisms (shrinking encircling mechanism and the spiral-shaped path) with probability of 50% during the optimization process. So that if p <0,5 the shrinking encircling is used to update the position, else the spiral-shaped path will be used.

Whales also have a certain probability of searching for prey when they are constructing bubble-network. Mathematically, searching a prey enhance WOA exploration, This phase is based on the change of *A* coefficient. If *A* exceeds the range of [−1, 1], the distance $\vec{D}$ is updated randomly. At this time, whales will deviate from the original optimal fitness, so that the algorithm has a certain global search-ability, which is formulated as follow:

$$\vec{D} = \left|\vec{C}.\overrightarrow{X_{\text{rand}}} - \vec{X}\right| \qquad (9)$$

$$\vec{X}(t+1) = \overrightarrow{X_{\text{rand}}} - \vec{A}.\vec{D} \qquad (10)$$

Where, $\overrightarrow{X_{rand}}$ is random location information of a whale selected from this iteration. The flowchart of WOA technique is depicted in Algorithm **1**.

Algorithm 1. Whale Optimization Algorithm

```
Input Number of MaxIter and Population etc
Initialize the whales population Xi (i = 1, 2, ..., n)
Initialize a, A, C, l and p
Calculate the fitness of each search agent
X*= the best search agent
while (it < MaxIter)
    for each search agent
        if (p < 0.5)
            if (|A| < 1)
                Update the position of the current search agent by the equation (4)
            else if (|A| ≥ 1)
                Select a random search agent (X_rand)
                Update the position of the current search agent by the equation (10)
            End
        else if (p ≥ 0.5)
            Update the position of the current search by the by the equation (8)
        End
    End
    Calculate the fitness of each search agent
    Update X* if there is a better solution
    it=it+1
    Update a, A, C, l and p
end while
return X*
```

4. PROPOSED METHOD

Our proposal achieves a noval miner for rule miner based on a whale optimization algorithm named WO-ARM. It aims to extract the most trustworthy association rules in less time and less computational needs. In this section, the used database layout, encoding, and fitness function are highlighted. Furthermore, the modified whale algorithm will be described.

A. Database layout

Database presentation has a real effect on time and resource consumption due to a large volume of sales in such a database Also, the number of scans over the any algorithm execution will influence directly on execution time. Generally, transnational database can be represented in horizontal, vertical and bitmap representation [38]. Therefore, the vertical layout was chosen for our approach, Because item X's support is the tidset dimension, also, To calculate the item-set A{X, Y} support, Tid-set of A needs to be defined as the intersection of X Tid-set and Y Tid-set. Fig.2 shows an instance of layout transformation from horizontal to vertical layout.

B. Rule Encoding

In the literature, several representations of the rule exist to mine association rules using genetic algorithms or meta-heuristic algorithms.

T_id	item
1	A,B,C
2	A,C
3	B,C

(A) HORIZONTAL LAYOUT

T_id	A	B	C
1	1	1	1
2	1	0	1
3	0	1	1

(B) VERTICAL LAYOUT

Figure. 2. Database representation

The main two, as discussed in [39], are Pittsburgh and Michigan, where the first consider a a set of solutions as one chromosome, whereas, within the second, each chromosome indicates one solution. Our work opts for the last codification. In which we code each rule (solution) X as a vector contains *J* items, whereas, *J* presents the items number in the data-set. Where the victor is coded as follows:

- S[i] = 1 if the i^{th} item exist in the *if statement* (antecedent),
- S[i] = 2 if the i^{th} item exist the *then statement* (consequence),
- S[i] = 0 when the item will not appear in the rule.

For instance: let I = {i_1, i_2, ..., i_{10} } be a set of items : the rule i1 , i5 ⇒ i6 , i2 , i7. is coded as X1 = {1, 2, 0, 0, 1, 2, 2, 0, 0, 0} Fig.3 shows a sample rule coded.

1	2	0	0	1	2	2	0	0	0

Figure. 3. Rule Encoding

C. fitness function

As reported in the background section, in the ARM issue, if support and confidence of such rule satisfy user threshold then the rule is accepted. The suggested algorithm aims to maximize the objective function, which supervises uppermost rules extraction. If α and β are two observational parameters utilized for weighting amid the utilized measures inside the objective function, which showed as follow:

$$f(r) = \begin{cases} \alpha.support + \beta.confedence & IfAcceptedRul \\ -1 & Otherwise \end{cases} \quad (11)$$

D. Algorithm description

Algorithm 2 illustrates the pseudo-code of WO-ARM. That has three principal stages. Which are outlined as follows:

- ***Data-set preprocessing:*** as mentioned earlier in this section, our proposal is based on a vertical data-set layout. In which less computational necessaries needed to calculate the support and confidence of each generated rule—also, no whole database scan to complete the

computation process. The central fact of this choice is shown in the time response; for these reasons, a preprocessing step in required to convert from horizontal to vertical representation.

- ***Parameter initialization:*** Firstly, all whales are initialized by arbitrary rules, and initial values attributed to all other whale algorithm parameters. After the fitness function for each rule is calculated, and the best rule is chosen and affected to *X**. This phase is entirely randomized.
- ***Rules extraction:*** The same concepts are reused from the original whale optimization algorithm[9], in which the authors use each whale as a candidate solution that will be improved toward the optimal solution. Our proposal assumes that each whale is a nominee rule that includes n elements, where n denotes Items number within the transnational database. The general process of obstetrics a new rule is based on changing items values in each rule based on whale optimization algorithm process (Encircling prey, Bubble-net attacking method, and Search for prey). Next, the rule will be validated to our encoding using a simple way based on odd or even number, such as: if the Item is odd, it belongs to the rule antecedent, else if it is even it belongs to the rule consequence. Otherwise, the Item is not in the rule, which will be 0. Afterword, the algorithm calculates the fitness for each whale and replaces the optimal solution X* by the new one, and this search process will be reiterated until the maximum number of iterations is attained.

E. Algorithm complexity

According to algorithm 2, WO-ARM has a simple structure that is similar to WOA. According to ARM problem which is an NP-hard problem[40]. There is a little difference in the WO-ARM algorithm complexity by adding a number of Items changed in a rule which represents a solution. Therefore, in worst case the WO-ARM complexity is *O(n* Max_iterations*2*J)*, where *n* is the whales number, Max_iterations is the iterations number and J is the Item number in the transactional database.

5. EXPERIMENTATION AND RESULTS

In order to demonstrate the performance of our method, we did various tests on the recommended algorithm, WO-ARM. This part describes the utilized datasets. Afterword, a comparison recently developed similar approaches is provided.

A. Benchmark and setup description

In order to evaluate our proposal performance, the study utilized various data-sets,which are famous and commonly real-world in data mining, in many tests, taken from Frequent and mining data-set Repository[41], Bilkent University Function Approximation Repository[42].

Algorithm 2. Whale optimization algorithm for association rule mining

```
0:  Data-set preprocessing
1:  Input Number of MaxIter and Population, minsupport, minconfedance
2:  Initialize the population Xi (i = 1, 2, ..., n)
3:  Initialize a, A, C, L and p
4:  Compute the fitness function of each search whale
5:  X*= the best rule
6:  While(t<=MaxIter)
7:      Update a, A, C, L, and p
8:      For all whales in the population do
9:          If (p<0.5)
10:             If (|A|<1)
11:                 For each Item in the solution Xi
12:                 Update Item by using equation (4)
13:             Else if(|A|=1)
14:                 Select a random Item in Xi
15:                 Update Item using equation (10)
16:             End if
17:         Else if (p ≥ 0.5)
18:             For each Item in the solution Xi
19:             Update Item by using equation (8)
21:         End if
22:         For each Item in the solution Xi
23:             If the Item is odd, it belongs to the antecedent,
                Otherwise, it belongs to the consequence
24:     End for
25:     Calculate the fitness of each search agent
26:     Update X* if there is a better solution
27:     it=it+1
28: End while
29: Return X*
```

Table.1 shows the different datasets included in our tests. The data-sets vary from one to the other in terms of transaction size, item number, and the overage items number per transaction. As example, Chess data-set includes 3196 activities with 75 elements, while each transaction contains an average of 37 items, unlike the mushroom data-set, which more significant in terms of transactions and items, whereas it has just 23 items per transaction.

This section describes the datasets and tests setup. after, the outcomes achieved will be given. The last section will show a comparative report beside diverse

advanced optimization techniques in the field of ARM. Moreover, a comparative study to exact algorithms in terms of memory consumption will be illustrated.

Note: All algorithms in our study are written in JAVA and all tests were conducted on a machine Intel core I5 with 4Go ram running on Linux Ubuntu.

TABLE 1. DESCRIPTION OF EXPERIMENTAL BENCHMARK

Data-set	Transactions size	Item size	Average size
Basketball	96	5	8
Bodyfat	252	15	8
IBM - standard	1,000	20	20
Quak	2,178	4	5
Chess	3,196	37	37
Mushroom	8,124	23	23

B. Stability study

In this section, we focus on the stability of our proposal (WO-ARM). In other words, we study how deals WO-ARM with objective function and CPU time in terms of redundancies. On the other hand, how deals with whales number changing with objective function and CPU time. These tests aim to extract the best parameters (Number of population and Maximum number iteration), that can reach the best results of our algorithm. In this study, we used four datasets with an average size of transactions, which are IBM-standard, Quak, Chess, and Mushroom. We execute whale optimization algorithm for ARM 20 times, and the average results are taken, the support and confidence thresholds fixed to 0.2 and 0.5, respectively.

TABLE 2. PERFORMANCE OF THE WO-ARM WITH DIFFERENT NUMBERS OF ITERATIONS (TIME IN SECOND)

#Itr	IBM-STD		Quak		Chess		Mushroom	
	Time	Fitness	Time	Fitness	Time	Fitness	Time	Fitness
100	0,5	1	0,6	0,55	1	0,91	5,2	0,74
200	1,2	1	1	0,72	3	0,83	10	0,72
300	1,6	1	1,7	0,73	4,7	0,89	19	0,87
400	1,9	1	2,6	0,86	6,8	0,89	24	0,86
500	2,2	1	2,9	0,99	7,9	0,88	27	0,97
600	3,4	1	4,5	0,98	11,2	0,92	31	0,98
700	3,7	1	4,7	0,99	12,5	0,96	38	0,98
800	3,9	1	4,9	1	13,9	0,96	42	0,98
900	4,1	1	5,5	1	16,5	0,95	47	0,99
1000	4,4	1	6,8	1	19,3	0,96	63	1

Table.2 shows the outcomes obtained by our tests. That present the performance of the proposed algorithm WO-ARM in terms of the iterations number, which is changed. Change regularly from 100 to 1000 iteration. These results are obtained with a fixed number of whales to 30 whales. In terms of fitness, we can observe that our proposal achieves its best results at 500 iterations. With different datasets except for IBM-Quest-standard in which the best result was obtained at 100 iterations. On the other hand, we can note that CPU time growing with the iterations increment, which is a natural behavior of each swarm-based algorithm.

On the opposite side, the number of whales in the population also influences the stability of the algorithm. With this in mind, we repeated our tests. Whereas, this time by fixing the maximum iterations number, and change the agents' number in the population, that changed from 10 to 50 regularly. Table. 3 illustrates the results achieved by our algorithm. From the outcome, it is noted that the best fitness in Quack, Chess, and Mushroom datasets is achieved within 30 whales and after this number, almost the same fitness function is obtained. With IBM-Quest-standard dataset the best fitness obtained with ten whales because it has the smallest number of transactions which makes it simpler in exploration than other datasets.

TABLE 3. PERFORMANCE OF THE WO-ARM WITH DIFFERENT NUMBERS OF WHALES (TIME IN SECOND)

#pop	IBM-STD		Quak		Chess		Mushroom	
	Time	Fitness	Time	Fitness	Time	Fitness	Time	Fitness
10	0,6	1	0,7	0,68	2,8	0,61	7	0,84
20	1,7	1	2,2	0,94	4,2	0,89	20	0,90
30	2,3	1	2,8	0,99	8,9	0,90	27	0,97
40	3,7	1	4,5	0,99	14,6	0,91	39	0,98
50	5,5	1	5,9	1	16,5	0,91	69	1

These promising results in terms of rules quality can be explained the good trad-off between exploration and exploitation in the WOA which uses a shrinking encircling mechanism and the spiral-shaped path mechanisms in exploitation and searches the prey for exploration of the search space.

C. Comparative study to other approaches

To well place our method against other methods designed in the literature for ARM. In this section, we present a comparative study that focuses on CPU time, rules quality, and memory consumption. This comparison divided into two main steps, firstly we compare our method in-face-of single-objective optimization approaches in terms of CPU time and rule quality, and secondly, the WO-ARM compared to exact methods in terms of memory consumption. To make this comparison fair, we use the same machine for all algorithms and use the best parameter for each one. For WO-ARM we fixed the maximum number of iterations to 500 and whales number to 30.

1) In-face-of single-objective optimization techniques

The results of WO-ARM were analysed facing to the following four well-known algorithms:

- Penguins Search Optimization Algorithm for Association Rules Mining (Pe-ARM) [26],
- Bees swarm optimization algorithm for ARM (BSO-ARM) [7],
- Multi-swarm bat algorithm for ARM (MSB-ARM) [29],
- Bat algorithm for Association Rule Mining (BAT-ARM) [8].

Results show the average of 20 execution for each algorithm. Table.4 illustrates the out-comes of each algorithm in terms of time consumption with six datasets with different sizes. It is observed that WO-ARM outperforms the other algorithms with the majority of datasets. Except with mushroom where BSO-ARM has less runtime compared to WO-ARM within 0,9 seconds which can be negligible. These outcomes can be explained by the whale optimization algorithm complexity, which inherited by whale optimization algorithm for association rule mining (WO-ARM).

TABLE 4. COMPARING OUR APPROACH TO EXISTING APPROACHES W.R.T TIME (SECOND)

	Pe-ARM	BSO-ARM	MSB-ARM	BAT-ARM	WO-ARM
Basketball	1,5	3,36	4	7	**0,7**
Bodyfat	2,88	5,7	11	14	**1,3**
IBM-std	1,68	1,92	13	19	**1,2**
Quak	3,35	4,5	40	76	**2,3**
Chess	4,92	5,1	13	141	**4,7**
Mushroom	10,68	**9,1**	144	341	10

Indeed, the runtime is not enough to judge such a swarm-inspired algorithm. Solution quality is an essential factor that influences on decisions as a good algorithm or not. With this in mind, we compare our proposal in-face-of mentioned above algorithms in terms of fitness function value, and the outcomes are illustrated in Table.5.

Again, WO-ARM proves its superiority against other algorithms with the majority of datasets which can be explained by the excellent trad-off between intensification and diversification in the whale optimization algorithm. Also, thanks to the shrinking encircling mechanism and the spiral-shaped path mechanisms that guide the algorithms to choose the best neighbor in local search.

2) In-face-of exact methods methods

As highlighted in the introduction, one of the most challenging drawbacks in exact algorithms for association rule mining is memory consumption. Especially with the vast stored data our days. To overcome this drawback, optimization algorithms are investigated for ARM in order to discover WO-ARM memory usage.

TABLE 5. COMPARING OUR APPROACH TO EXISTING APPROACHES W.R.T FITNESS

	Pe-ARM	BSO-ARM	MSB-ARM	BAT-ARM	WO-ARM
Basketball	1	0,92	1	0,81	1
Bodyfat	1	0,73	1	0,54	1
IBM-std	0,92	0,93	0,84	0,41	**0,94**
Quak	0,91	1	1	0,52	1
Chess	0,89	0,88	0,97	0,92	**0,99**
Mushroom	0,88	0,75	0,68	0,93	**0,97**

The algorithm tested with four datasets. Moreover, the outcomes are compared to those from exact algorithms (Apriori, FP-growth) and result obtained from the Multi-swarm bat algorithm for ARM (MSB-ARM), and Bat algorithm for Association Rule Mining (BAT-ARM). The results are summarized in Table. 6. The results presents that WO-ARM has less memory usage compared to other algorithms. These results are related firstly to dataset representation which gives less computation in the phase of rule evaluation. Also, whale optimization algorithm complexity influences memory usage.

TABLE 6. COMPARING OUR APPROACH TO EXACT APPROACHES W.R.T MEMORY USAGE (MB)

	Apriori	Fp-growth	MSB-ARM	BAT-ARM	WO-ARM
IBM-std	26,05	26,22	27,2	13,74	**17,6**
Quak	19,12	25,12	21,6	**16,2**	18,01
Chess	225,33	104,55	58,32	48,63	**31,46**
Mushroom	317,58	291,1	256,7	170,29	**52,62**

6. CONCLUSION

Nowadays, Association rules have widely used to define relationships between items in databases. Nevertheless, association rule mining is an NP-complete problem; time and memory consumption are explosively grown with the number of transactions and items in the database, making rule extraction a challenging problem for exact algorithms. To overcome this challenge, this paper presented a whale optimization algorithm for association rule mining (WO-ARM). In which, we investigated in the good trad-off between intensification and diversification that distinguished the original whale optimization algorithm based on shrinking encircling mechanism, the spiral-shaped path, and search prey technique. We evaluated the proposed algorithm on six well-known datasets in the field of ARM, and the outcomes are compared to recently developed similar

approaches. Results showed the effectiveness of WO-ARM in terms of runtime, quality, and memory consumption. These results are obtained due to the whale optimization algorithm mechanisms. In the near future, we aim to develop our proposition to handle large scale datasets. The improvement will be concertized by the use of parallel execution on Graphical Processing Units (GPU).

REFERENCES

[1] W. J. Frawley, G. Piatetsky-Shapiro, and C. J. Matheus, "Knowledge discovery in databases: An overview," *AI Mag.*, vol. 13, no. 3, p. 57, 1992.

[2] R. Agrawal, T. Imieliński, and A. Swami, "Mining association rules between sets of items in large databases," in *ACM SIGMOD Record*, 1993, vol. 22, no. 2, pp. 207–216.

[3] R. Agrawal, R. Srikant, and others, "Fast algorithms for mining association rules," in *Proc. 20th int. conf. very large data bases, VLDB*, 1994, vol. 1215, pp. 487–499.

[4] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," in *ACM SIGMOD Record*, 2000, vol. 29, no. 2, pp. 1–12.

[5] W. Wang and S. M. Bridges, "Genetic Algorithm Optimization of Membership Functions for Mining Fuzzy Association Rules," 2000.

[6] Z. Kou and L. Xi, "Binary Particle Swarm Optimization-Based Association Rule Mining for Discovering Relationships between Machine Capabilities and Product Features," 2018, doi: 10.1155/2018/2456010.

[7] Y. Djenouri, H. Drias, and Z. Habbas, "Bees swarm optimisation using multiple strategies for association rule mining," *Int. J. Bio-Inspired Comput.*, vol. 6, no. 4, pp. 239–249, 2014.

[8] K. E. Heraguemi, N. Kamel, and H. Drias, "Association Rule Mining Based on Bat Algorithm," *J. Comput. Theor. Nanosci.*, vol. 12, no. 7, pp. 1195–1200, 2015.

[9] S. Mirjalili and A. Lewis, "The Whale Optimization Algorithm," *Adv. Eng. Softw.*, vol. 95, pp. 51–67, May 2016, doi: 10.1016/j.advengsoft.2016.01.008.

[10] F. S. Gharehchopogh and H. Gholizadeh, "A comprehensive survey: Whale Optimization Algorithm and its applications," *Swarm Evol. Comput.*, vol. 48, no. November 2018, pp. 1–24, 2019, doi: 10.1016/j.swevo.2019.03.004.

[11] G. Nalcaci and M. ERMİŞ, "Selective Harmonic Elimination for Three-Phase Voltage Source Inverters Using Whale Optimizer Algorithm," Accessed: Jun. 21, 2020. [Online]. Available: https://avesis.metu.edu.tr/yayin/dd544ccd-4fe9-4180-8ba0-7938df0b3b78/selective-harmonic-elimination-for-three-phase-voltage-source-inverters-using-whale-optimizer-algorithm.

[12] R. K. Saidala and N. R. Devarakonda, "Bubble-net hunting strategy of whales based optimized feature selection for e-mail classification," in *2017 2nd International Conference for Convergence in Technology, I2CT 2017*, Dec. 2017, vol. 2017-January, pp. 626–631, doi: 10.1109/I2CT.2017.8226205.

[13] J. Nasiri and F. M. Khiyabani, "A whale optimization algorithm (WOA) approach for clustering," *Cogent Math. Stat.*, vol. 5, no. 1, Jun. 2018, doi: 10.1080/25742558.2018.1483565.

[14] S. J. Mousavirad and H. Ebrahimpour-Komleh, "Multilevel image thresholding using entropy of histogram and recently developed population-based metaheuristic algorithms," *Evol. Intell.*, vol. 10, no. 1–2, pp. 45–75, Jul. 2017, doi: 10.1007/s12065-017-0152-y.

[15] M. J. Zaki, "Scalable algorithms for association mining," *Knowl. Data Eng. IEEE Trans.*, vol. 12, no. 3, pp. 372–390, 2000.

[16] M. J. Zaki and C.-J. Hsiao, "CHARM: An Efficient Algorithm for Closed Itemset Mining.," in *SDM*, 2002, vol. 2, pp. 457–473.

[17] J. Mata, J. L. Alvarez, and J. C. Riquelme, "Mining numeric association rules with genetic algorithms," in *Artificial Neural Nets and Genetic Algorithms*, 2001, pp. 264–267.

[18] R. Haldulakar and J. Agrawal, "Optimization of Association Rule Mining through Genetic Algorithm.," *Int. J. Comput. Sci. Eng.*, vol. 3, no. 3, 2011.

[19] X. Yan, C. Zhang, and S. Zhang, "Genetic algorithm-based strategy for identifying association rules without specifying actual minimum support," *Expert Syst. Appl.*, vol. 36, no. 2, pp. 3066–3076, 2009.

[20] S. Sarkar, A. Lohani, and J. Maiti, "Genetic algorithm-based association rule mining approach towards rule generation of occupational accidents," in *Communications in Computer and Information Science*, 2017, vol. 776, pp. 517–530, doi: 10.1007/978-981-10-6430-2_40.

[21] B. S. Neysiani, N. Soltani, R. Mofidi, and M. H. Nadimi-Shahraki, "Improve Performance of Association Rule-Based Collaborative Filtering Recommendation Systems using Genetic Algorithm," *Int. J. Inf. Technol. Comput. Sci.*, vol. 11, no. 2, pp. 48–55, 2019, doi: 10.5815/ijitcs.2019.02.06.

[22] P. Kumar and A. K. Singh, "Efficient Generation of Association Rules from Numeric Data Using Genetic Algorithm for Smart Cities," 2019, pp. 323–343.

[23] R. J. Kuo, C. M. Chao, and Y. T. Chiu, "Application of particle swarm optimization to association rule mining," *Appl. Soft Comput.*, vol. 11, no. 1, pp. 326–336, 2011.

[24] K. Sarath and V. Ravi, "Association rule mining using binary particle swarm optimization," *Eng. Appl. Artif. Intell.*, vol. 26, no. 8, pp. 1832–1840, 2013.

[25] Y. Djenouri, H. Drias, Z. Habbas, and H. Mosteghanemi, "Bees Swarm Optimization for Web Association Rule Mining," in *Web Intelligence and Intelligent Agent Technology (WI-IAT), 2012 IEEE/WIC/ACM International Conferences on*, 2012, vol. 3, pp. 142–146.

[26] Y. Gheraibia, A. Moussaoui, Y. Djenouri, S. Kabir, and P. Y. Yin, "Penguins Search Optimisation Algorithm for Association Rules Mining," *CIT. J. Comput. Inf. Technol.*, vol. 24, no. 2, pp. 165–179, 2016.

[27] X.-S. Yang, "A new metaheuristic bat-inspired algorithm," in *Nature inspired cooperative strategies for optimization (NICSO 2010)*, Springer, 2010, pp. 65–74.

[28] K. E. Heraguemi, N. Kamel, and H. Drias, "Multi-population Cooperative Bat Algorithm for Association Rule Mining," in *Computational Collective Intelligence*, Springer, 2015, pp. 265–274.

[29] K. E. Heraguemi, N. Kamel, and H. Drias, "Multi-swarm bat algorithm for association rule mining using multiple cooperative strategies," *Appl. Intell.*, vol. 45, no. 4, pp. 1021–1033, Dec. 2016, doi: 10.1007/s10489-016-0806-y.

[30] U. Mlakar, M. Zorman, and I. Fister, "Modified binary cuckoo search for association rule mining," *J. Intell. Fuzzy Syst.*, vol. 32, pp. 4319–4330, 2017, doi: 10.3233/JIFS-16963.

[31] S. M. Ghafari and C. Tjortjis, "A survey on association rules mining using heuristics," *WIREs Data Min. Knowl. Discov.*, vol. 9, no. 4, Jul. 2019, doi: 10.1002/widm.1307.

[32] P. Ganghishetti and R. Vadlamani, "Association Rule Mining via Evolutionary Multi-objective Optimization," in *Multi-disciplinary Trends in Artificial Intelligence*, Springer, 2014, pp. 35–46.

[33] M. M. J. Kabir, S. Xu, B. H. Kang, and Z. Zhao, "A New Evolutionary Algorithm for Extracting a Reduced Set of Interesting Association Rules," in *Neural Information Processing*, 2015, pp. 133–142.

[34] K. E. Heraguemi, N. Kamel, and H. Drias, "Multi-objective bat algorithm for mining numerical association rules," *Int. J. Bio-Inspired Comput.*, vol. 11, no. 4, pp. 239–248, 2018, doi: 10.1504/IJBIC.2018.092797.

[35] Z. Zhang, N. Chai, E. Ostrosi, and Y. Shang, "Extraction of association rules in the schematic design of product service system based on Pareto-MODGDFA," *Comput. Ind. Eng.*, vol. 129, pp. 392–403, Mar. 2019, doi: 10.1016/j.cie.2019.01.040.

[36] R. Agrawal, T. Imielinski, and A. Swami, "Mining Association Rules between Sets of Items in Large Databases."

[37] P.-N. Tan, V. Kumar, and J. Srivastava, "Selecting the right objective measure for association analysis," *Inf. Syst.*, vol. 29, no. 4, pp. 293–313, 2004.

[38] J. Han, J. Pei, and M. Kamber, *Data mining: concepts and techniques*. Elsevier, 2011.

[39] A. A. Freitas, *Data mining and knowledge discovery with evolutionary algorithms*. Springer, 2002.

[40] F. Angiulli, G. Ianni, and L. Palopoli, "On the Complexity of Mining Association Rules.," in *SEBD*, 2001, pp. 177–184.

[41] B. Goethls and M. J. Zaki, "Frequent Itemset Mining Dataset Repository," 2003, [Online]. Available: http://fimi.ua.ac.be/data/.

[42] H. A. Guvenir and I. Uysal, "Bilkent University Function Approximation Repository," 2000, [Online]. Available: http://funapp.cs.bilkent.edu.tr/DataSets/.

KamelEddine Heraguemi is is Director of digitization at M'sila University, and lecturer at Computer Science Department, Faculty of mathematics and computer science, Med Boudiaf University, M'sila, Algeria. He received his PhD in Computer Science from the University Ferhat Abbas of Setif1 (UFAS1), Setif and Master's in Computer Science from the Mohamed-Cherif Messaadia University – Souk-Ahras, Algeria, in 2017 and 2012, respectively. In 2013, he joined the research team Data Mining and Machine Learning at the Laboratory of Research in Artificial Intelligence (LRIA) at USTHB, Algeria. His research interests include data mining, artificial intelligence, and evolutionary computing.

Houria Kadri is a researcher in computer science, at Mouhamed Boudiaf University at M'sila, Algeria. She received her license degree in 2018 from the same university. Her research interests include data mining, artificial intelligence, and evolutionary computing.

Amira Zabi is a researcher in computer science, at Mouhamed Boudiaf University at M'sila, Algeria. She received her license degree in 2018 from the same university. Her research interests include data mining, artificial intelligence, and evolutionary computing.