Complex & Intelligent Systems (2021) 7:297–309
https://doi.org/10.1007/s40747-020-00200-0

ORIGINAL ARTICLE

A new recommendation system using map-reduce-based tournament empowered Whale optimization algorithm

Ashish Kumar Tripathi[1] · Himanshu Mittal[2] · Pranav Saxena[2] · Siddharth Gupta[2]

Received: 27 March 2020 / Accepted: 12 September 2020 / Published online: 27 September 2020

Abstract

In the era of Web 2.0, the data are growing immensely and is assisting E-commerce websites for better decision-making. Collaborative filtering, one of the prominent recommendation approaches, performs recommendation by finding similarity. However, this approach fails in managing large-scale datasets. To mitigate the same, an efficient map-reduce-based clustering recommendation system is presented. The proposed method uses a novel variant of the whale optimization algorithm, tournament selection empowered whale optimization algorithm, to attain the optimal clusters. The clustering efficiency of the proposed method is measured on four large-scale datasets in terms of F-measure and computation time. The experimental results are compared with state-of-the-art map-reduce-based clustering methods, namely map-reduce-based K-means, map-reduce-based bat algorithm, map-reduce-based Kmeans particle swarm optimization, map-reduce-based artificial bee colony, and map-reduce-based whale optimization algorithm. Furthermore, the proposed method is tested as a recommendation system on the publicly available movie-lens dataset. The performance validation is measured in terms of mean absolute error, precision and recall, over a different number of clusters. The experimental results assert that the proposed method is a permissive approach for the recommendation over large-scale datasets.

Keywords Recommendation system · Big data · Map-reduce · Clustering · Whale optimization algorithm

Introduction

Among the various web revolutions, recommendation system is a prominent tool which is widely used by E-commerce websites to offer more personalized services to the users. For example, movie recommendation method suggests a list of movies that a specific user may prefer based on the information retrieved from the social media or rating made by other similar users [1]. Generally, a recommendation system follows two types of approaches, namely content-based filtering and collaborative filtering. In content-based filtering, each item is associated with a certain set of features which are rated differently by different users. This approach predicts the rating of the items on the basis of user's inputs [2,3]. On the contrary, collaborative filtering takes up a completely different approach. It works on the similarity among the users or items [4]. The performance of such recommendation systems is highly dependent on the similarity determination. Generally, clustering-based approaches are quite popular in the literature to determine the similarity [5].

K-means, a widely used clustering approach, has been used in a number of engineering domains for the same. However, K-means generates biased clusters due to its dependence over parameter settings and initial cluster centres [6]. To remedy this concern, meta-heuristic-based solutions have been widely employed to obtain optimal cluster centroids in the last two decades [7–9]. Pal et al. [10] introduced a new clustering algorithm using the enhanced bio-geography algorithm. Furthermore, Mittal et al. [11] presented an intelligent gravitation search algorithm-based method to obtain optimal cluster centroids. Sharma et al. [12] introduced an enhanced grey wolf optimization-based method for the optimal clustering of the data. Pal et al. [13] presented genetic algorithm-based energy-efficient weighted clustering method. Recently, a number of researchers have used meta-heuristic-based clustering solutions for recommendation systems. Chen et al. [14] introduced collaborative filtering-based recommendation method using evolutionary clustering. Malik et al. [15] introduced particle swarm-based travel recommendation sys-

✉ Himanshu Mittal
himanshu.mittal224@gmail.com

[1] Malviya National Institute of Technology, Jaipur, India

[2] Jaypee Institute of Information Technology, Noida, India

tem. Moreover, Peška et al. [16] performed a detailed study about the applicability of meta-heuristic-based methods for solving the collaborative filtering-based recommendation system. Kumar et al. introduced efficient clustering-based model for the movie recommendations [17]. Kataria [18] introduced artificial bee colony-based movie recommendation system. Similarly, Singh et al. [19] introduced novel movie recommendation system by the efficient clustering of the dataset using modified cuckoo search method. Suganeshwari at al. [20] performed a survey on clustering-based recommendation system and concluded that clustering-based recommendation system can be efficiently utilized for the recommendations of the product and services as it finds the similarity among the the user behavior and uses patterns.

Generally, meta-heuristic methods optimize cluster centroids based on the inter-cluster or intra-cluster distances. Unlike K-means, these methods obtain the optimal solution through collective working, which eradicates any biasness towards initial clusters. Hence, these methods perform better for the clustering problem. Therefore, this paper presents a novel meta-heuristic-based recommendation system for the big data environment.

Meta-heuristic methods refer to the set of algorithms which leverages the concept of guided random search. These methods define a mathematical model which correspond to certain natural phenomena and have been used in the literature to obtain optimal solutions for different real-world optimization problems [21–25]. Generally, they use population-based approach to finds the optimal solution with the information sharing among the individuals. In contrast, single solution-based methods such as simulated annealing and hill climbing [26], finds the solution with a single individual. However, single solution-based algorithm suffers with premature convergence due to the lack of information sharing. Furthermore, the success of a meta-heuristic algorithm majorly depends on the way in which exploration and exploitation is performed [27,28]. Exploration controls the diversification of the search agents, whereas the convergence of the individuals is controlled by the exploitation. Therefore, each meta-heuristic method tries to attain balance between exploration and exploitation to achieve precise solution [29]. Generally, these algorithms are inspired from swarm-based, or evolution-based phenomenons. Mirjalili et al. [30] developed multi-verse algorithm based on the notion of cosmology. Sayed et al. [31] introduced hybrid SA-MFO algorithm solving the engineering design problems. The genetic algorithm, differential evolution and bio-geography-based optimization are some of the popular examples of evolutionary concept [32]. Furthermore, swarm-based algorithms behave like the swarm of agents to achieve optimal results. Particle swarm optimization (PSO) is one the meta-heuristic that has been broadly used solving problems and several variants of the PSO has also been introduced in the literature [33]. Subsequently, Unal et al. [34] presented multi-objective particle swarm optimization, which uses random immigrants. Lie et al. [35] introduced levy flight based ant colony optimization. Moreover, Satapathy [36] presented the social group optimization, which mimics the social behavior of humans for solving the problems. Furthermore, Tripathi et al. [37] proposed an algorithm inspired by military dog squad to find the optimal solution. Dragonfly-based optimization is another swarm-based algorithm introduced by Mirjalili et al. [38].

WOA [39] is a popular algorithm which models the behavior of humpback whales. Mathematically, WOA simulates the hunting behavior of whales to find the optimal solution. It includes two phases, namely encircling phase and spiral phase, which corresponds to exploration and exploitation, respectively. WOA has surpassed other recent algorithms on the benchmark problems [39]. In the last three years, WOA has been applied across a wide set of application areas, like data clustering, mining, image processing, and others [40]. Moreover, WOA has been improved by several researchers for solving various real-world problems. Mafarja et al. [41] introduced hybrid WOA and simulated annealing-based method for the feature selection. Aziz et al. [42] combined moth fame algorithm with WOA for the multi-level image segmentation. Similarly, Aljarah et al. [43] employed WOA for optimizing connection weights of the neural network. Furthermore, the whale algorithm has also performed competitive in the recommendation system. Karleka et al. [44] introduced a WOA-based clinical risk assessment and recommendation method for treatment. However, collaborative filtering-based recommendation method involves clustering of data according to user's similarity. Moreover, literature has witnessed that WOA performs efficiently in clustering-based applications [45]. Therefore, this paper aims at leveraging the strengths of WOA for collaborative-filtering-based recommendation system.

Generally, WOA discards bad solutions during position updation. However, the whale having bad fitness might be nearer to global optima [41]. Therefore, it suffers from demerits like the risk of trapping into local optima [46]. To remedy this, a new variant of WOA, tournament selection empowered WOA (TWOA), is proposed in this paper. The tournament process gives a fair chance to the bad solutions to overcome the local optima during exploitation. Furthermore, the strength of TWOA is utilized for improving the quality of the recommendation system. Although meta-heuristic-based recommendation system has shown better efficiency than traditional methods comparatively, these sequentially executing recommendation systems fail to respond in a reasonable amount of time on large-scale datasets [47]. To alleviate the same, the TWOA is parallelized using the map-reduce architecture for large-scale datasets and has been leveraged to obtain optimal clusters to perform recommendations.

The overall contribution of this paper is two folds, (1) a new clustering method, map-reduce-based tournament empowered whale optimization algorithm (MR-TWOA), is presented for efficient clustering of large-scale data set and (2) a novel variant of the WOA, tournament empowered whale optimization algorithm (TWOA), is presented to attain efficient clustering. The clustering efficiency of the proposed map-reduce-based TWOA (MR-TWOA) is tested on four large datasets, namely Replicated Iris, Replicated CMC, Replicated Wine, and Replicated Vowel. The experimental findings are compared with other state-of-the-art map-reduce-based clustering methods, namely map-reduce-based K-means (MR-Kmeans) [7], map-reduce-based bat algorithm (MR-bat) [48], map-reduce-based Kmeans particle swarm optimization (MR-KPSO) [49], map-reduce-based artificial bee colony (MR-ABC) [50], and map-reduce-based whale optimization (MR-WOA). Furthermore, the applicability of the proposed MR-TWOA-based recommendation system is validated using MovieLens dataset [51]. The results are compared with three parameters, namely mean absolute error (MAE), precision, and recall.

The remaining sections of the paper are as follows. In this section, briefs data-clustering and WOA. The next section discusses the proposed recommendation system along with the proposed variant (TWOA) and its parallel version (MR-TWOA). The Experimental results section presents the experimental arrangements and results. Finally, the paper is concluded in the last section.

Preliminaries

Clustering

Data clustering is an unsupervised machine learning approach which iteratively groups the set of N data-points in p clusters. Unlike supervised approaches, it does not need any priori training phase. Let $O = \{0_{11}, o_{12}, \ldots, o_{1t}\}$, $\{o_{21}, o_{22}, \ldots, o_{2t}\}$, and $\{o_{n1}, z_{n2}, \ldots, o_{nt}\}$ be a set of n data-points having t features and o_{ij} denotes the jth attribute value of ith data-point. The clustering works iteratively to find a set of cluster centroids denoted as $K = \{k_{11}, k_{12}, \ldots, k_{1t}\}$, $\{k_{21}, k_{22}, \ldots, k_{2t}\}$, and $\{k_{p1}, k_{p2}, \ldots, k_{pt}\}$. k_{ij} corresponds to the value of jth attribute of ith cluster centroid and $k_i = k_{i1}, k_{i2}, \ldots, k_{it}$ is the position vector for ith cluster-centroid. Generally, the intra-cluster distance is considered as the objective function while performing clustering which is defined as the Euclidean distance between O_i and K_l. Its formulation is depicted in Eq. (1).

$$(O_i, K_l) = \sum_{l=1}^{p} \sum_{i=1}^{n} \left(\sqrt{\sum_{t=1}^{t} \left(O_i^t - K_l^t\right)^2} \right) \tag{1}$$

where O_i and K_l represent ith data-point and lth cluster, respectively.

Whale optimization algorithm (WOA)

Whale optimization algorithm [39] mimics the hunting behavior of humpback whales. The humpback whales hunt small fishes in the proximity surface by generating bubbles in a circular shape. The algorithm works in the two phases, namely exploration and exploitation. Furthermore, the exploitation phase is performed through two different strategies, namely shrinking encircling and spiral update. In shrinking encircling mechanism, the whale moves toward the best whale in a circular manner.

Exploitation phase

To mathematically model exploitation phase of WOA, current best is represented by the position of the prey, which is assumed as the solution nearest to the optimum solution. To exploit the search space, the position of each whale is defined according to the prey, which simulated as encircling behavior. The current position of each agent is defined using two ways, namely spiral formation and encircling of prey. The encircling of prey is equated as Eq. (2).

$$\mathbf{P}(m+1) = \mathbf{P}_b(m) - \mathbf{A} \cdot D \tag{2}$$

where position $\mathbf{P}(m)$, denotes the position of agent at iteration m and $\mathbf{P}_b(m)$ represents the best agent. $\mathbf{A}$ represents the coefficient vector which is equated in Eq. (3) while D denotes the distance from best agent which is computed as Eq. (4).

$$A = 2\mathbf{a} \cdot \mathbf{r} - \mathbf{a} \tag{3}$$

$$D = \mid \mathbf{C} \cdot \mathbf{X}_b(m) - \mathbf{X}(m) \mid \tag{4}$$

$$\mathbf{C} = 2 \cdot \mathbf{r} \tag{5}$$

where $r \in (0, 1)$ is a randomly generated number, a is linearly decreasing vector with values from 2 to 0, and $\mathbf{C}$ denotes an adjustment factor by which search agents captures the local areas.

Furthermore, the spiral formation is mathematically modeled as Eq. (6).

$$\mathbf{X}(m+1) = \acute{D} \cdot e^{bl} \cdot cos(2\pi l) + \mathbf{X}_b(m) \tag{6}$$

where l represents is a randomly generated number in the range $[-1, 1]$, constant number b defines spiral shape, and $(\acute{D})$ represents the distance between prey and search agent as defied in Eq. (7).

$$\acute{D} = \mathbf{X}_b(m) - \mathbf{X}(m) \tag{7}$$

The exploitation phase of the WOA is implemented with equal probabilities using Eq. (8).

$$\mathbf{P}(i+1) = \begin{cases} \text{Encircling phase Eq. (2)} & q < 0.5 \\ \text{Spiral phase Eq. (6)} & q \geq 0.5 \end{cases} \tag{8}$$

here $q \in (0, 1)$ is randomly generated number.

Exploration phase

To perform the exploration, each whale updates its position either randomly in the search space or using the best search agent, which depends on vector **A**.

For $\mathbf{A} > 1$, a random movement is performed by whales whereas for $\mathbf{A} < 1$, whales prefer to search locally in the space. The exploration phase is mathematically modelled as Eqs. (9) and (10) at iteration $(t + 1)$.

$$\mathbf{D} = \mid \mathbf{C} \cdot \mathbf{P}_{\text{rand}} - \mathbf{P}(t) \mid \tag{9}$$

$$\mathbf{P}(m+1) = \mathbf{P}_{\text{rand}} - \mathbf{A} \cdot \mathbf{D} \tag{10}$$

where $\mathbf{P}_{rand}$ denotes any randomly selected whale. Algorithm 1 details the pseudo-code of the WOA.

Proposed method

This section details a novel recommendation system, namely map-reduce-based tournament empowered WOA (MR-TWOA), to deal with large-scale data efficiently. The proposed method performs clustering by leveraging the strengths of map-reduce architecture with TWOA. The workflow of the MR-TWOA is depicted in Fig. 1. First, the user-rated dataset is captured. Then, it is processed through the proposed MR-TWOA to obtain optimal clusters in an efficient manner.

Algorithm 1 Whale optimization algorithm (WOA) [39]

Input: Population (P_j) randomly generated in search space, $j := 1, 2, \ldots, n$
Output: P^* (final position of best whale i.e prey)
Find the fitness of each whale and position of prey (P^*)
while ($it < Iter_{max}$) **do**
  for each whale in the population **do**
    Update l, p, **A**, **C**, and **a**
    if ($q < 0.5$) **then**
      if $\mid A \mid < 1$ **then**
        Redefine the positions of whale using encircling phase
      else if $\mid A \mid \geq 1$ **then**
        Initialize (X_{rand})
        Redefine the position of whale using exploration phase
      end if
    else if ($q \geq 0.5$) **then**
      Update the positions of whale using spiral phase
    end if
  end for
  perform boundary checks
  Find the fitness of each whale and prey (P^*)
  it := it+1
end while
Return P^*

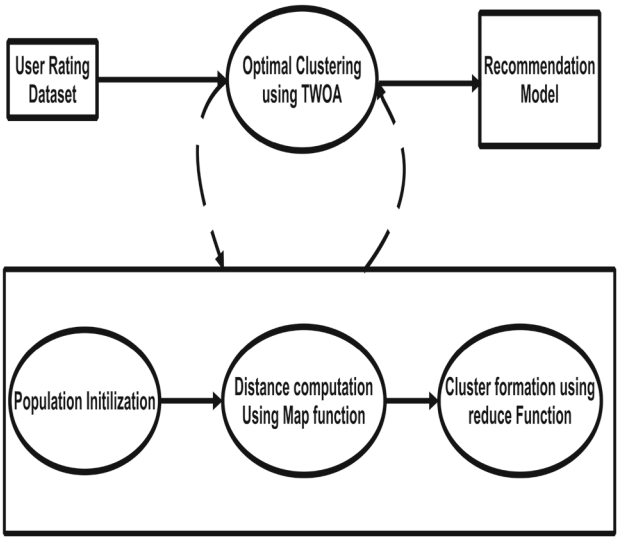


Fig. 1 The proposed Map-reduce-based tournament empowered WOA for recommendation

Here, each whale corresponds to a set of cluster centroids which are defined over d dimensions, where d corresponds to the number of features in the considered dataset. The similarity measure among the user-rating is considered as the clustering criteria. Finally, recommendations are made to the users based on the obtained clusters. In the following section, the proposed variant (TWOA) is detailed, followed by the parallel version of MR-TWOA for clustering the large-scale dataset.

Tournament empowered WOA

WOA defines the position of the optimal solution according to the current best whale and randomly selected whale. The parameter 'a' controls the equilibrium between exploration and exploitation. However, WOA performs exploration using the randomly picked solution, which affects the exploration and exploitation balance. To mitigate the above concern, a novel tournament selection empowered WOA has been introduced. Instead of a random solution in the exploration phase, TWOA uses tournament selection [52] for selecting the $\mathbf{P}_{\text{rand}}$ solution in Eqs. (9) and (10). This yields a better possibility of selecting good solutions at the later stage. This results in fast convergence and better exploitation.

MR-TWOA-based recommendation method

For clustering using meta-heuristic algorithm, each iteration involves $N * K * P$ number of distance computations, where N denotes the number of data points, K is the number of clusters, and P denotes the population size. Therefore, on large scale datasets, sequential algorithms fail to respond in terms

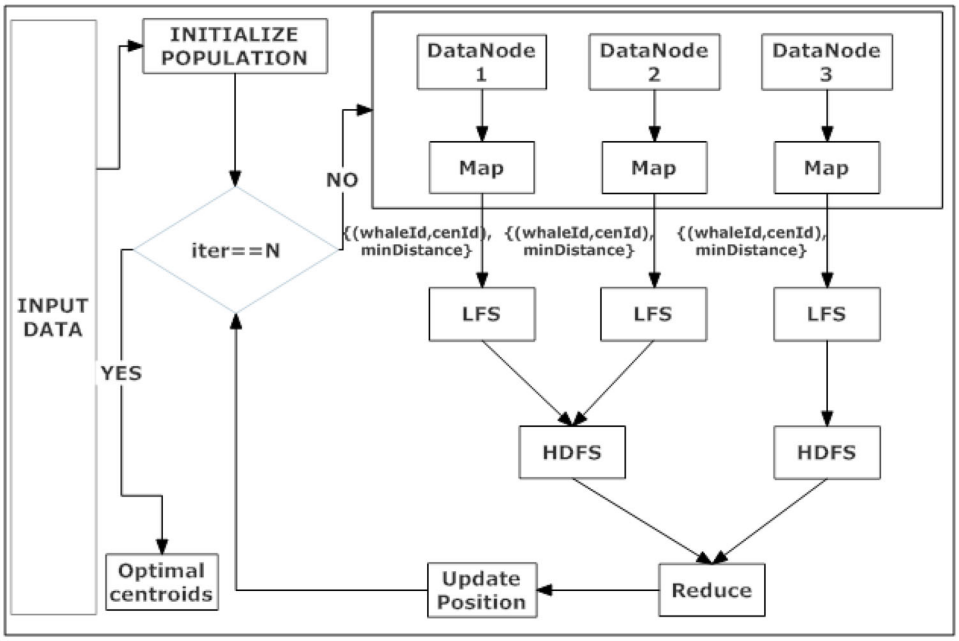


Fig. 2 The map-reduce architecture of MR-TWOA

Algorithm 2 Map Phase of MR-TWOA

Input: Map (Key: objectId, Value: Object) **Output:** centroid and distance of each data point F
Initialization:
key=objectID
value=Object
read(WOA-population from file);
for each whale **in** WOA-population;
whaleID =retrieve-whaleID(whalePopulation)
centroidList =retrieve-centroids(woaPopulation) /* the position of each whale denotes centroids location*/
Distance= getMinimum(object, centroidList);
/* The getMinimum() function returns the minimum distance as explained below*/
Initialization:
centroid-ID=read-centroidList() /* to get the position of first centroid*/
minimum-Distance=getDistance(object, centroidList)
for each cluster centroid-ID **do**
  dis = get distance of j^{th} centroid from data object
  if $(dis < minimum - Distance)$ **then**
    minimum-Distance=dis
    centroid-ID = i / *i represents index of the centroid list with least distance */
  end if
end for
new-key= whaleID+centroidID;
end for
emit (new-key, minimum-Distance);

of memory and computation time. To remedy this, a parallel model of TWOA named as MR-TWOA is presented using Hadoop architecture based on MapReduce. Particularly, MR-TWOA runs over a cluster of computers in which data-points are distributed uniformly among the Hadoop distributed file system (HDFS). The complete architecture of MR-TWOA is presented in Fig. 2. As shown in the figure, the large dataset is first broken into small size input splits (64 MB). For the first iteration, the MR-TWOA population is randomly initialized within the search boundaries. Furthermore, the population file is sent to each mapper running on the cluster. In the proposed recommendation system, the computation of the sum of the squared Euclidean distance (fitness value) is required at each iteration, which takes the majority of the computation cost. Therefore, this task of fitness calculation is parallelized using the mapper function of the Map-Reduce. The proposed MR-TWOA works in two phases, namely MR-TWOA-Map and MR-TWOA-Reduce. MR-TWOA-Map clusters the data-points and finds clusters, with the clustering criteria as the least Euclidean distance between the data-point and corresponding centroid. The pseudo-code of the map phase is detailed in Algorithm 2. As depicted in Algorithm 2, the MR-TWOA-Map phase first retrieves the cluster centers from the population stored in the HDFS. After that, the minimum with distance each data object is calculated with the centroids. The outcome of the this phase is {key:(whaleId,cenId), value:minDistance}, where '$whaleId$' denotes the identification of whale for clusters matched with the data-point and '$cenId$' represents the identification of cluster-centroid with minimum distance from the data-point. '$minDistance$' is the Euclidean distance between data-point and the centroid with identification '$cenId$'. After the completion of the Map phase, the output from all the mappers is collected and grouped by the key. Then, MR-TWOA-Reduce phase processes the distances obtained in Map phase and calculates the intra-cluster distance for each centroid, defined for each whale. The outcome of this phase is of the form {key :

Algorithm 3 Reduce Phase of MR-TWOA

Input:{key:(whaleId,cenId), value:minimumDistance}
Output: The final position of centroid, fitness value
Initialization:
C=0
total-Distance=0
for each instance in the value list **do**
c++ / *c records the counter value for finding mean */
total-Distance+=value
end for
new-key=redefined position of whales as per new fitness
$Emit(new-key, total-Distance)$

$(whaleId, cenId)$, $value : intra-clusterdistance\}$. The pseudo-code of Reduce phase is illustrated in Algorithm 3.

Time complexity

The time complexity of MR-TWOA-based recommendation method is proportional to the number of clusters, the number of data objects, and the number of dimensions in the dataset. In the MR-TWOA based recommendation method, the optimal number of centroids are obtained with $O(N \times C \times D \times T)$ operations, where N, C, D, and T denotes the total number of data objects, number of clusters and number of dimensions in the dataset, and number of iterations, respectively. Furthermore, for the population size of P, the time complexity of the proposed recommendation system can be represented as $O(P \times N \times C \times D \times T)$.

Experimental results

The performance of MR-TWOA method is analyzed in three sections. First, the efficacy of the proposed TWOA is validated on 23 benchmarks which belong to three different categories, namely uni-modal, multi-modal, and fixed dimensional multi-modal. Second, the clustering efficiency of the parallel version of TWOA (MR-TWOA) has been analyzed on four large-scale datasets. In the third section, the experimental validation of the proposed method (MR-TWOA) as the recommendation system is performed in terms of three parameters, namely mean absolute error (MAE), recall, and precision.

Performance of TWOA on benchmark problems

This section details the experimental analysis of the proposed variant (TWOA) on 23 standard benchmark functions. The simulation results are conducted on a computer having Intel Corei3-4570 processor with 3.20 GHz, 4GB ram and 500 GB hard disk. The results are compared with four recent meta-heuristic methods, namely whale optimization algorithm (WOA) [39], improved cuckoo search (ICS) [53], enhanced grey-wolf optimizer (EGWO) [12], and salp-swarm algorithm (SSA) [54]. As WOA has already shown superior performance over popular meta-heuristic methods in literature such as grey wolf optimizer [55], particle swarm optimization (PSO) [56], dragonfly algorithm [38], differential evolution [57]. Therefore, the comparison includes only recently proposed meta-heuristic methods. Tables 1, 2, 3 detail the considered 23 benchmark functions which are grouped into three categories, namely unimodal, multi-modal, and fixed dimensional multi-modal functions, respectively. Generally, unimodal functions describe the exploitation ability of the considered method, while multi-modal functions validate the exploration ability of the method. Furthermore, each method is executed over 30 times for each benchmark function. The best fitness value obtained in different runs is averaged and analyzed in terms of mean fitness value and standard deviation. The parameter settings of each meta-heuristic method are given in Table 4. These values were fixed according to the related literature to make a fair comparison between the selected meta-heuristics [12,39,53,54]. Moreover, the population size and the number of iterations for all algorithms are kept as 30 and 500, respectively.

Table 1 Description of unimodal benchmark functions

Function	V_{no}	Range	f_{min}
$F_1(x) = \sum_{i=1}^{n} x_i^2$	30	[− 100, 100]	0
$F_2(x) = \sum_{i=1}^{n} \mid x_i \mid + \prod_{i=1}^{n} \mid x_i$	30	[− 10, 10]	0
$F_3(x) = \sum_{i=1}^{d} \sum_{j=1}^{i} x_j^2$	30	[− 100, 100]	0
$F_4(x) = \max_i \{\mid x \mid, 1 \le i \le n\}$	30	[− 100, 100]	0
$F_5(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	30	[− 30, 30]	0
$F_6(x) = \sum_{i=1}^{n} ([x_i + 0.5])^2$	30	[− 100, 100]	0
$F_7(x) = \sum_{i=1}^{n} i x_i^4 + random[0, 1)$	30	[− 1.28, 1.28]	0

Table 5 tabulates the average fitness value on different benchmark functions obtained by the considered meta-heuristic methods along with the standard deviation. It is pertinent from the table that TWOA outperforms the other compared methods on four unimodal functions, i.e. F_1, F_2, F_5, $F7$. For F_3 and F_4. ICS has shown competitive results while SCA performed well on F_6. Thus, it may be stated that TWOA has superior local searchability. Moreover, TWOA has surpassed other methods on more than 80% of the multimodel functions. This represents that TWOA is robust against trapping in local optima. The superiority of TWOA is due to the inclusion of the tournament selection process which resulted in better trade-off between the exploration and exploitation. Additionally, the poor solutions also got a fair chance in the early phase of the algorithm, which prevents the algorithm from the premature convergence.

Table 2 Description of multi-modal benchmark functions

Function	V_{no}	Range	f_{min}
$F_8(x) = \sum_{i=1}^{n} -x_i \sin\sqrt{\lvert x_i \rvert}$	30	[− 500, 500]	0
$F_9(x) = \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i) + 10]$	30	[− 5.12, 5.12]	0
$F_{10}(x) = -20\exp^{-0.02\sqrt{n^{-1}\sum_{i=1}^{n} x_i^2}} - e^{n^{-1}\sum_{i=1}^{n} cos(2\pi x_i)} + 20 + e$	30	[− 32, 32]	0
$F_{11}(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos(\frac{x_i}{\sqrt{i}}) + 1$	30	[− 600, 600]	0
$F_{12}(x) = \frac{\pi}{n}\left\{10\sin(\pi y_1) + \sum_{i=1}^{n-1}(y_i - 1)^2\left[1 + 10\sin^2(\pi y_1)\right] + (y_n - 1)^2\right\}$ $+ \sum_{i=1}^{n} u(x_i, 10, 100, 4),\ y_i = 1 + \frac{x_i+1}{4},$ $u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i > 0 \\ 0 & -a < x_i < 1 \\ k(-x_i - a)^m & -x_i - a \end{cases}$	30	[− 50, 50]	0
$F_{13}(x) = 0.1\{\sin^2(3\pi x_1) + \sum_{i=1}^{n}(x_i - 1)\left[1 + \sin^2(3\pi x_i + 1)\right]$ $+(x_n - 1)^2\left[1 + \sin^2(2\pi x_n)\right]\} + \sum_{i=1}^{n} u(x_i, 5, 100, 4)$	30	[− 50, 50]	0

Table 3 Description of fixed-dimension multi-modal benchmark functions

Function	V_{no}	Range	f_{min}
$F_{14}(x) = \left(\frac{1}{500} + \sum_{j=1}^{25}\frac{1}{j + \sum_{i=1}^{2}(x_i - a_{ij})^6}\right)^{-1}$	2	[− 65, 65]	1
$F_{15}(x) = \sum_{i=1}^{11}\left[a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4}\right]^2$	4	[− 5,5]	0.0003
$F_{16}(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 + 4x_2^2 + 4x_2^4$	2	[− 5,5]	− 1.0316
$F_{17}(x) = (x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6)^2 + 10(1 - \frac{1}{8\pi})\cos x_1 + 10$	2	[− 5, 5]	0.398
$F_{18}(x) = \left[1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)\right]$ $\times\left[30 + (2x_1 + 3x_2 + 1)^2(18 - 32x_1 + 12x_1^2 - 48x_2 + 36x_1x_2 + 27x_2^2)\right]$	2	[−2, 2]	3
$F_{19}(x) = -\sum_{i=1}^{4} c_i \exp(-\sum_{j=1}^{3} a_{ij}(x_j - p_{ij})^2)$	3	[1, 3]	− 3.86
$F_{20}(x) = -\sum_{i=1}^{4} c_i \exp(-\sum_{j=1}^{6} a_{ij}(x_j - p_{ij})^2)$	6	[0, 1]	− 3.32
$F_{21}(x) = -\sum_{i=1}^{5}\left[(X - a_i)(X - a_i)^T + c_i\right]^{-1}$	4	[0, 10]	− 10.1532
$F_{22}(x) = -\sum_{i=1}^{7}\left[(X - a_i)(X - a_i)^T + c_i\right]^{-1}$	4	[0, 10]	− 10.4028
$F_{23}(x) = -\sum_{i=1}^{1} 0[(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0, 10]	− 10.5363

Table 4 Parameter setting of the TOWA and other considered algorithms

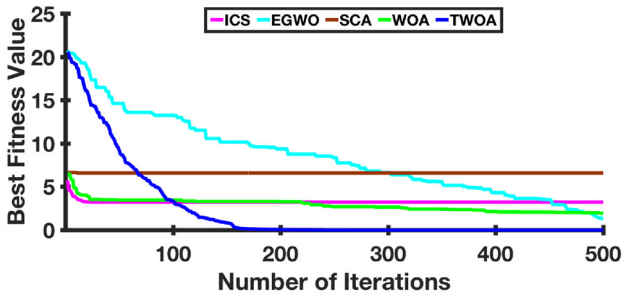
Parameter name	SCA	ICS	EGWO	WOA	TWOA
Population size (pop)	30	30	30	30	30
Number of iterations (itr)	500	500	500	500	500
a	2	2	2	2	2
Probability (Pa)	–	.25	–	–	–
Step scaling factor	–	.01	–	–	–
Crossover rate (C)	–	0.1	–	–	–
Mutation rate (C)	–	–	0.1	–	–

Furthermore, to analyze the exploration and exploitation behavior, the convergence trends of the proposed and considered methods on two representative benchmark functions, namely F_1 and F_8, are depicted in Fig. 3. In the figure, the horizontal axis represents the iteration count, and vertical axis denotes the best fitness value. It is visualizable from convergence curves that TWOA smoothly reaches the optimal solution. This shows that the proposed method has better ability to attain an optimal solution. Therefore, it can be validates from experimental analysis that TWOA is an efficient method that can be leveraged for clustering the large scale datasets.

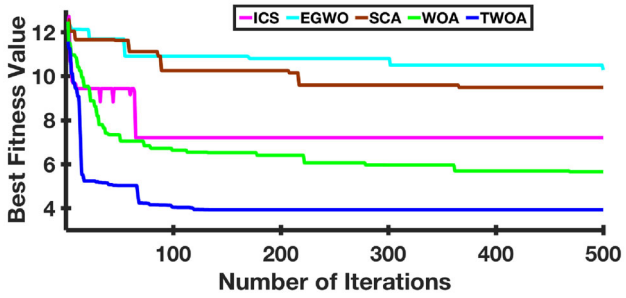
Table 5 Mean and standard deviation of the fitness value over 30 runs

Fn.	TWOA		WOA		ICS		EGWO		SCA	
	MEAN	STD	MEAN	STD	MEAN	STD	MEAN	STD	MEAN	STD
F1	**7.04E−73**	**2.71E−72**	8.71E−73	4.40E−72	4.56E+01	2.38E+01	1.64E+01	3.09E+01	2.32E−03	3.02E−03
F2	**4.14E−51**	**7.47E−51**	1.04E−50	1.21E−50	2.50E+01	3.11E+01	3.00E−02	4.51E−02	5.09E+01	5.63E+01
F3	9.19E+03	5.58E+03	6.22E+04	9.06E+03	**4.45E+03**	**2.49E+03**	1.11E+04	6.66E+03	6.10E+03	3.73E+03
F4	3.62E+01	1.97E+01	6.39E+01	3.19E+01	**1.88E+01**	1.60E+01	4.39E+01	**4.64E+00**	2.07E+01	4.92E+00
F5	**2.76E+01**	**3.08E−01**	3.45E+01	5.64E−01	9.51E+03	2.02E+04	1.78E+05	7.77E+05	3.04E+02	4.72E+02
F6	1.06E+00	2.93E−01	5.22E−01	3.35E−01	4.66E+01	2.58E+01	2.21E+01	2.82E+01	**1.72E−03**	**1.73E−03**
F7	**1.10E−04**	**7.02E−05**	2.82E−03	3.05E−03	4.71E−02	1.77E−02	1.05E−01	1.17E−01	3.22E−01	1.14E−01
F8	**− 1.32E+04**	**1.28E+02**	− 1.09E+04	1.92E+03	− 9.06E+03	7.45E+02	− 4.53E+03	3.70E+02	− 7.08E+03	9.09E+02
F9	1.07E+02	2.13E+01	**2.31E−15**	**1.19E−14**	1.36E+02	3.76E+01	4.25E+01	3.97E+01	1.05E+02	3.20E+01
F10	**8.75E−16**	**8.11E−01**	5.42E−15	3.38E+01	6.74E+00	1.43E+00	1.96E+01	1.03E+01	6.23E+00	3.16E+00
F11	**6.77E−02**	**3.43E−02**	1.83E−01	8.79E+00	1.45E+00	1.87E−01	1.18E+00	3.36E−01	8.90E−02	1.06E−01
F12	**1.67E−02**	**7.68E−03**	6.16E−02	2.86E−01	1.28E+01	6.24E+00	1.82E+05	1.05E+06	1.71E+01	6.76E+00
F13	**5.70E−01**	**2.11E−01**	7.66E−01	4.25E−01	6.76E+01	1.18E+02	9.36E+04	2.60E+05	3.50E+01	2.61E+01
F14	3.75E+00	2.92E+00	3.73E+00	3.84E+00	**1.25E+00**	**4.87E−16**	2.38E+00	2.57E+00	3.04E+00	2.15E+00
F15	**7.17E−04**	**2.25E-04**	1.05E−03	1.29E−03	8.14E−03	1.92E−02	1.29E−03	4.65E−04	4.62E−03	8.64E−03
F16	**− 1.33E+00**	**7.76E−14**	− 1.26E+00	1.55E−09	− 1.29E+00	5.18E−13	− 1.23E+00	7.01E−05	− 1.02E+00	5.14E−05
F17	**3.91E−01**	**2.43E−09**	4.86E−01	3.09E−05	4.98E−01	3.43E−08	4.76E−01	1.33E−08	5.15E−01	1.93E−07
F18	**2.96E+00**	**5.53E−13**	3.66E+00	1.58E−04	7.13E+00	1.84E+01	3.59E+00	1.55E−04	3.89E+00	4.48E−03
F19	**− 4.99E+00**	**2.85E−13**	− 4.70E+00	1.35E−02	− 4.73E+00	2.47E−01	− 4.60E+00	2.92E−03	− 3.80E+00	2.06E−02
F20	**−4.18E+00**	**4.03E−02**	− 3.89E+00	2.60E−01	− 4.06E+00	8.06E−02	− 3.59E+00	1.96E−01	− 3.23E+00	1.41E−01
F21	− 7.53E+00	4.36E+00	− 3.14E+00	**1.57E+00**	− 7.59E+00	3.33E+00	**− 8.97E+00**	2.45E+00	− 7.26E+00	3.21E+00
F22	**− 9.65E+00**	**1.89E+00**	− 9.02E+00	3.61E+00	− 6.48E+00	4.41E+00	− 3.11E+00	2.38E+00	− 7.38E+00	4.19E+00
F23	**− 8.56E+00**	**2.19E+00**	− 8.54E+00	3.69E+00	− 7.30E+00	4.66E+00	− 4.46E+00	2.27E+00	− 7.54E+00	4.60E+00

Bold represents best value



(a) F_1



(b) F_8

Fig. 3 Convergence trend of TWOA with other considered meta-heuristics

Performance analysis of MR-TWOA

To test the clustering efficiency of MR-TWOA, four extremely large datasets are considered, namely replicated CMC, replicated Vowel, replicated Iris, replicated Wine. The datasets are formed by replicating each sample of the original dataset 1000 times. Table 6 contains the detailed description of the considered datasets. The clustering efficiency of the proposed MR-TWOA is measured in terms of F-measure and computation time. Furthermore, the MR-TWOA clustering results are compared against four recent map-reduce-based clustering methods, namely map-reduce-based K-means (MR-Kmeans), map-reduce-based bat algorithm (MR-BAT), map-reduce-based Kmeans particle swarm optimization (MR-KPSO), map-reduce-based artificial bee colony (MR-ABC), and map-reduce based whale optimization algorithm (MR-WOA). The parallelization of the clustering method is achieved through Hadoop 2.6.2 on Ubuntu 14.04 operating system and simulated in java 1.8.0. Table 7 presents the F-measure (Fm) and computation time (CT) of the considered methods in terms of mean value which is obtained over 30 runs by running the considered methods on a cluster of 5 computers. It is visible from the table that MR-TWOA has outperformed the compared methods on all datasets. The performance of MR-Kmeans algorithm has been recorded as poorest among all the considered methods. However, it has given competitive performance in terms of computation time since it works on single solution-based approach.

Table 6 Description of the considered large datasets

Name	Cluster	Dimension	Data objects
Iris (Replicated)	3	7	10,000,050
CMC (Replicated)	3	9	10,000,197
Wine (Replicated)	2	18	5,000,000
Vovel (Replicated)	10	10	1,025,010

Moreover, the parallel computation efficacy of MR-TWOA is validated in terms of speedup which is computed according to Eq. (11).

$$S = T_{\text{base}}/T_{\text{N}} \tag{11}$$

where T_{base} represents the computation time taken by a method to run on a single machine, and T_{N} refers to the time taken by the same method to run on N number of machines. To study the speedup efficiency of MR-TWOA, two large-scale datasets are considered, namely Replicated Iris and Replicated CMC. Figure 4a and b represent the speedup graphs of MR-TWOA for Replicated Iris and Replicated CMC datasets, respectively. In the speedup graph, Y axis corresponds to the computation time while X axis corresponds to the number of machines (or nodes) in the cluster. From the figures, it is observable that the speedup performance of MR-TWOA running on Replicated Iris dataset is 2.7548 when there are five nodes in the cluster. The speedup performance of MR-TWOA running on Replicated CMC dataset is 2.1561 when there are five nodes in the cluster. This clearly indicates that MR-TWOA is an efficient method and can be used for large-scale clustering datasets.

Analysis of MR-TWOA as recommender system

This section analysis the applicability of the proposed MR-TWOA for the recommendation. To perform the same, MovieLens dataset [51] is considered which is a publicly available dataset, consisting of 1000 user-reviews on 1700 movies. It contains 100,000 data-points, where each data point corresponds to a user-rating for a movie. Furthermore, this dataset is replicated 1000 times to make it suitable for Hadoop architecture. To analyze the efficacy of the MR-TWOA with the considered map-reduce-based clustering methods, three performance measures, namely mean absolute error (MEA), precision, and recall, are considered over the different number of clusters. Table 8 depicts the MAE, precision, and recall of the considered methods. For the visual interpretation of Table 8, Figs. 5, 6, and 7 depict the bar-charts corresponding to mean absolute error, precision, and recall, respectively. The X axis in the figures corresponds to

Table 7 Computation time (CT) and F-measure (Fm) for 30 runs of the MR-TWOA and other methods

S. no	Dataset	Criteria	MR-Kmeans	MR-KPSO	MR-ABC	MR-BAT	MR-WOA	MR-TWOA
1	Repriduced Iris	Fm	0.636	0.767	0.833	0.781	0.801	**0.848**
		CT	**7.95E+04**	10.25E+04	10.27E+04	10.39E+04	10.20E+04	9.20E+04
2	Reproduced CMC	Fm	0.290	0.320	0.380	0.381	0.297	**0.392**
		CT	**7.80E+E04**	11.40E+E04	11.46E+04	10.41E+E04	11.52E+E04	10.51E+E04
3	Reproduced Wine	Fm	0.45	0.510	0.730	0.718	0.750	**0.790**
		CT	**10.12E+04**	17.19E+04	17.28E+04	20.29E+04	17.14E+04	16.15E+04
4	Reproduced Vovel	Fm	0.555	0.630	0.635	0.621	0.610	**0.650**
		CT	**11.65E+04**	15.32E+04	14.32E+04	14.20E+04	14.22E+04	13.26E+04

Bold represents best value

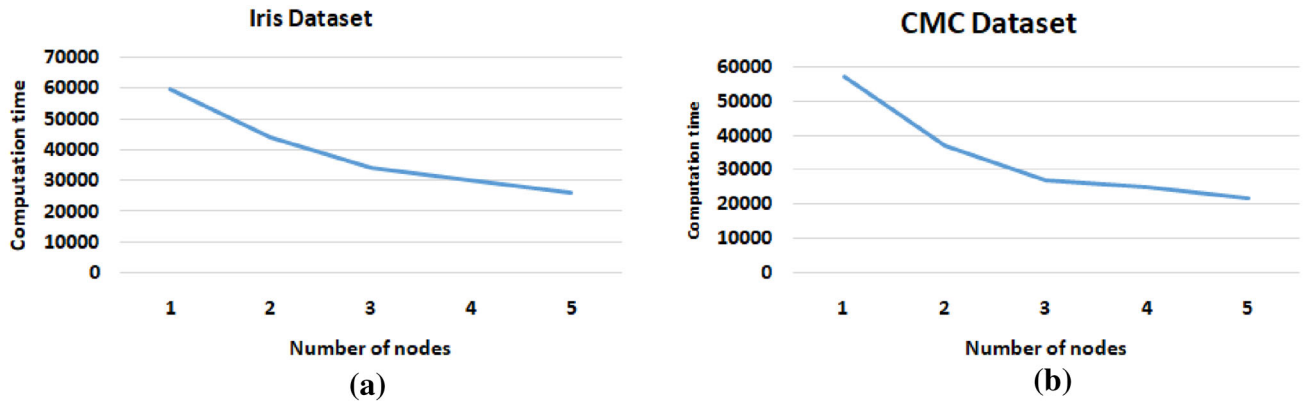


Fig. 4 Computation time analysis of MR-TWOA with other considered meta-heuristics

Table 8 Comparative analysis of MR-TWOA and other considered map-reduce-based clustering methods over different number of clusters

Clusters		5	10	15	20	25	30	35	40
MR-TWOA	MAE	0.741	0.690	0.681	0.690	0.689	0.680	0.686	0.687
	Precison	0.410	0.420	0.430	0.430	0.420	0.440	0.440	0.450
	Recall	0.130	0.120	0.210	0.370	0.450	0.550	0.670	0.690
MR-WOA	MAE	0.790	0.770	0.770	0.770	0.781	0.785	0.786	0.788
	Precision	0.410	0.370	0.390	0.360	0.350	0.390	0.360	0.350
	Recall	0.120	0.120	0.210	0.310	0.420	0.530	0.610	0.690
MR-BAT	MAE	0.820	0.790	0.790	0.790	0.800	0.805	0.806	0.807
	Precision	0.370	0.370	0.390	0.370	0.350	0.350	0.340	0.330
	Recall	0.130	0.110	0.260	0.270	0.300	0.380	0.440	0.700
MR-ABC	MAE	0.819	0.810	0.810	0.810	0.810	0.805	0.810	0.810
	Precision	0.350	0.320	0.330	0.320	0.320	0.320	0.310	0.320
	Recall	0.120	0.160	0.220	0.260	0.360	0.420	0.440	0.490
MR-PSO	MAE	0.825	0.825	0.824	0.828	0.824	0.824	0.825	0.825
	Precision	0.320	0.310	0.280	0.300	0.290	0.290	0.290	0.260
	Recall	0.100	0.120	0.160	0.230	0.340	0.440	0.460	0.500

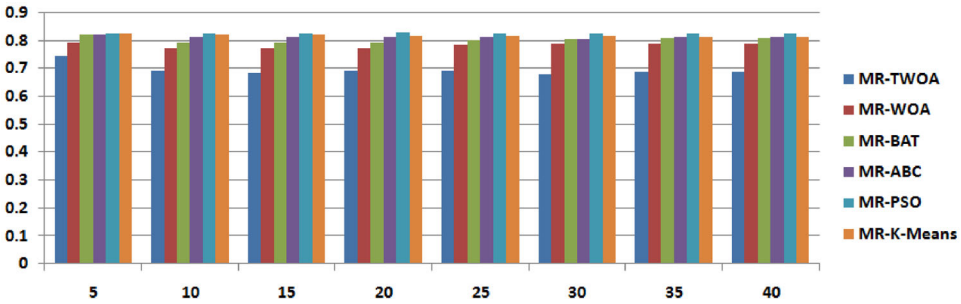


Fig. 5 Mean absolute error of MR-TWOA and other considered methods

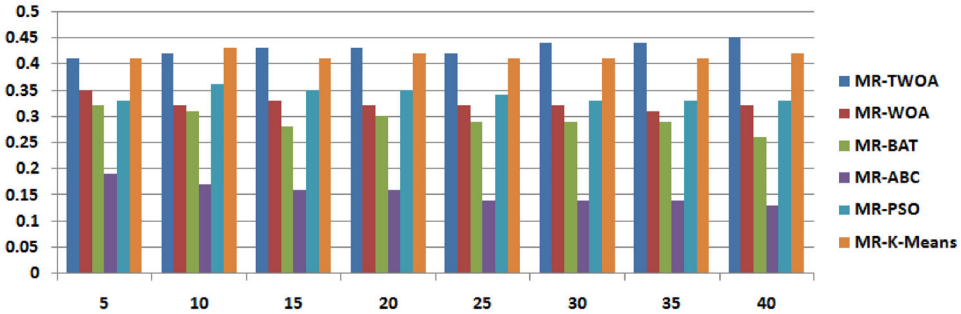


Fig. 6 Precision of MR-TWOA and other considered methods

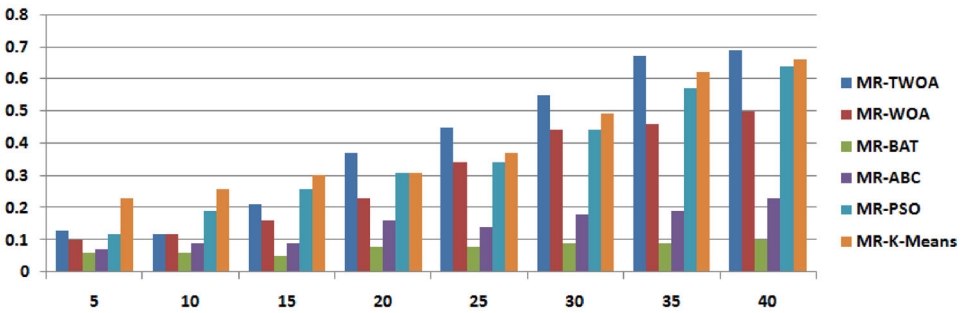


Fig. 7 Recall of MR-TWOA and other considered methods

the number of clusters, and *Y* axis represents the values of the considered measure. From the table and figures, it is visible that MR-TWOA has reported least MEA value among WOA, Bat, ABC and PSO on all the clusters. Whereas, WOA attained second least MEA all the clusters. Furthermore, it can also be observed that MR-TWOA has clearly outperformed all the methods in terms of precision. Again, WOA performed as second best method in terms of precision on all the clusters. It can also be inferred that MR-TWOA attains maximum recall among all the considered methods on all the cluster sets except 10, 15, where MR-BAT and MR-ABC has given competitive results, respectively. Furthermore, WOA has given second-best result when the number of clusters is set as 15, 20, 25, 30 and 40, while MR-Bat and ABC performed second best on 5 and 10 cluster sets, respectively. Therefore, it is affirmed from the experimental results that MR-TWOA is scalable and robust for data clustering. Moreover, it can be leveraged as a powerful alternative for the recommendation system over large-scale datasets.

Conclusion

In this paper, a novel recommendation method, MR-TWOA, is introduced for handling large dataset. The proposed method performs clustering through a novel variant of WOA, termed as tournament empowered WOA (TWOA). The performance of TWOA is tested on 23 uni-model and multi-model benchmark functions in terms of the mean and standard deviation of the fitness value. The results are compared against four recent meta-heuristic methods, namely WOA, ICS, EGWO, and SSA. The experimental results witnessed the superiority of the proposed method as compared to the considered methods on the majority of the benchmark function, which validates the ability of the TWOA for avoiding local optima. Furthermore, the clustering accuracy of the proposed MR-TWOA is tested on four massive datasets in terms of F-measure and computation time. The performance is compared with five recent map-reduce algorithms, namely MR-Kmeans, MR-KPSO, MR-ABC, MR-Bat, and MR-WOA. The proposed MR-TWOA outperformed the compared method on all the datasets, which shows the

superior clustering efficiency of the proposed method. Additionally, the performance of MR-TWOA is studied for the parallel environment in terms of speed-up efficiency. To do so, MR-TWO runs on a cluster with 5 machines for four massive datasets. The experimental results of the proposed MR-TWOA surpassed the other state-of-the-art metaheuristics-based methods. Furthermore, the recommendation ability of MR-TWOA is validated on MovieLens dataset in terms of MEA, precision and recall. It is confirmed from the simulation results that MR-TWOA outperformed the other considered methods in the product recommendation along with the ability to handle massive datasets.

In future, MR-TWOA can be used to unfold other real-world problems pertaining to big datasets. The proposed TWOA incorporates tournament selection for opting better solutions rather than random solutions. Since tournament selection sometimes fails in the selection of best solutions [58], it may limit the exploration ability of the proposed TWOA which can be improved by examining other selection methods. Furthermore, some other framework such as spark may be used to improve the computation cost of the proposed method.

References

1. Fu S, Yan Q, Feng GC (2018) Who will attract you? Similarity effect among users on online purchase intention of movie tickets in the social shopping context. Int J Inf Manag 40:88–102
2. Pazzani MJ, Billsus D (2007) Content-based recommendation systems. Tn: The adaptive web, Springer, pp 325–341
3. Yi S, Liu X (2020) Machine learning based customer sentiment analysis for recommending shoppers, shops based on customers? review. Complex Intell Syst 2020:1–14
4. Ahmadi A, Mukherjee D, Ruhe G (2019) A recommendation system for emergency mobile applications using context attributes: Remac. In: Proceedings of the 3rd ACM SIGSOFT international workshop on app market analytics, ACM, pp 1–7
5. Mittal H, Saraswat M (2020) A new fuzzy cluster validity index for hyper-ellipsoid or hyper-spherical shape close clusters with distant centroids. IEEE Trans Fuzzy Syst 2020:1–1. https://doi.org/10.1109/TFUZZ.2020.3016339
6. Mittal H, Saraswat M (2019) Classification of histopathological images through bag-of-visual-words and gravitational search algorithm. In: Lect. notes of soft computing for problem solving, Springer, pp 231–241
7. Zhao W, Ma H, He Q (2009) Parallel k-means clustering based on mapreduce. In: Cloud computing, Springer, pp 674–679
8. Katarya R, Verma OP (2018) Recommender system with grey wolf optimizer and FCM. Neural Comput Appl 30(5):1679–1687
9. Mittal H, Saraswat M (2018) An image segmentation method using logarithmic kbest gravitational search algorithm based superpixel clustering. Evolut Intell 2018:1–13
10. Pal R, Pandey HMA, Saraswat M (2016) Beecp: biogeography optimization-based energy efficient clustering protocol for hwsns. In: Contemporary Computing (IC3), 2016 Ninth International Conference on, IEEE, pp 1–6
11. Mittal H, Saraswat M (2019) An automatic nuclei segmentation method using intelligent gravitational search algorithm based superpixel clustering. Swarm Evolution Comput 45:15–32
12. Tripathi AK, Sharma K, Bala M (2018) A novel clustering method using enhanced grey wolf optimizer and mapreduce. Big Data Res 14:93–100
13. Pal R, Yadav S, Karnwal R et al (2020) Eewc: energy-efficient weighted clustering method based on genetic algorithm for hwsns. Complex Intell Syst 2020:1–10
14. Chen J, Zhao C, Chen L et al (2019) Collaborative filtering recommendation algorithm based on user correlation and evolutionary clustering. Complex Intell Syst 2019:1–10
15. Malik S, Kim D (2019) Optimal travel route recommendation mechanism based on neural networks and particle swarm optimization for efficient tourism using tourist vehicular data. Sustainability 11(12):3357
16. Peška L, Tashu TM, Horváth T (2019) Swarm intelligence techniques in recommender systems—a review of recent research. Swarm Evolution Comput 48:201–219
17. Kumar MS, Prabhu J (2020) A hybrid model collaborative movie recommendation system using k-means clustering with ant colony optimisation. Int J Internet Technol Secured Trans 10(3):337–354
18. Katarya R (2018) Movie recommender system with metaheuristic artificial bee. Neural Comput Appl 30(6):1983–1990
19. Singh SP, Solanki S (2019) A movie recommender system using modified cuckoo search. In: Emerging research in electronics, computer science and technology, Springer, pp 471–482
20. Suganeshwari G, Ibrahim SS (2016) A survey on collaborative filtering based recommendation system, In: Proceedings of the 3rd international symposium on big data and cloud computing challenges (ISBCC–16?), Springer, pp 503–518
21. Pandey AC, Rajpoot DS, Saraswat M (2017) Twitter sentiment analysis using hybrid cuckoo search method. Inf Process Manag 53:764–779
22. Pal R, Saraswat M (2019) Histopathological image classification using enhanced bag-of-feature with spiral biogeography-based optimization. Appl Intell 49(9):3406–3424
23. Mittal H, Saraswat M, Pal R (2020) Histopathological image classification by optimized neural network using igsa. In: International conference on distributed computing and internet technology, Springer, pp 429–436
24. Gupta V, Singh A, Sharma K, Mittal H (2018) A novel differential evolution test case optimisation (detco) technique for branch coverage fault detection. In: Smart computing and informatics, Springer, pp 245–254
25. Mittal H, Saraswat M (2018) ckgsa based fuzzy clustering method for image segmentation of rgb-d images. In: Proc. of international conference on contemporary computing, IEEE, pp 1–6
26. Selim SZ, Alsultan K (1991) A simulated annealing algorithm for the clustering problem. Pattern Recogn 24(10):1003–1008
27. Jaiswal K, Mittal H, Kukreja S (2017) Randomized grey wolf optimizer (rgwo) with randomly weighted coefficients. In: 2017 tenth

international conference on contemporary computing (IC3), IEEE, pp 1–3
28. Mittal H, Pal R, Kulhari A, Saraswat M (2016) Chaotic kbest gravitational search algorithm (ckgsa). In: Contemporary computing (IC3), 2016 Ninth international conference on IEEE, pp 1–6
29. Mittal H, Saraswat M (2018) An optimum multi-level image thresholding segmentation using non-local means 2d histogram and exponential kbest gravitational search algorithm. Eng Appl Artif Intell 71:226–235
30. Mirjalili S, Mirjalili SM, Hatamlou A (2016) Multi-verse optimizer: a nature-inspired algorithm for global optimization. Neural Comput Appl 27(2):495–513
31. Sayed GI, Hassanien AE (2018) A hybrid sa-mfo algorithm for function optimization and engineering design problems. Complex Intell Syst 4(3):195–212
32. Tripathi AK, Sharma K, Bala M (2019) Parallel hybrid bbo search method for twitter sentiment analysis of large scale datasets using mapreduce. Int J Inf Secur Privacy (IJISP) 13(3):106–122
33. Cheng S, Lu H, Lei X, Shi Y (2018) A quarter century of particle swarm optimization. Complex Intell Syst 4(3):227–239
34. Ünal AN, Kayakutlu G (2020) Multi-objective particle swarm optimization with random immigrants. Complex Intell Syst 2020:1–16
35. Liu Y, Cao B, Li H (2020) Improving ant colony optimization algorithm with epsilon greedy and levy flight. JSP 24(25):54
36. Satapathy S, Naik A (2016) Social group optimization (sgo): a new population evolutionary optimization technique. Complex Intell Syst 2(3):173–203
37. Tripathi AK, Sharma K, Bala M, Kumar A, Menon VG, Bashir AK (2020) A parallel military dog based algorithm for clustering big data in cognitive industrial internet of things. IEEE Trans Ind Informatics https://doi.org/10.1109/TII.2020.2995680
38. Mirjalili S (2016) Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. Neural Comput Appl 27(4):1053–1073
39. Mirjalili S, Lewis A (2016) The whale optimization algorithm. Adv Eng Softw 95:51–67
40. Medjahed SA, Saadi TA, Benyettou A, Ouali M (2016) Gray wolf optimizer for hyperspectral band selection. Appl Soft Comput 40:178–186
41. Mafarja MM, Mirjalili S (2017) Hybrid whale optimization algorithm with simulated annealing for feature selection. Neurocomputing 260:302–312
42. El Aziz MA, Ewees AA, Hassanien AE (2017) Whale optimization algorithm and moth-flame optimization for multilevel thresholding image segmentation. Expert Syst Appl 83:242–256
43. Aljarah I, Faris H, Mirjalili S (2018) Optimizing connection weights in neural networks using the whale optimization algorithm. Soft Comput 22(1):1–15
44. Karlekar NP, Gomathi N (2018) Ow-svm: Ontology and whale optimization-based support vector machine for privacy-preserved medical data classification in cloud. Int J Commun Syst 31(12):e3700
45. Nasiri J, Khiyabani FM (2018) A whale optimization algorithm (woa) approach for clustering. Cogent Math Stat 5(1):1483565
46. Ling Y, Zhou Y, Luo Q (2017) Lévy flight trajectory-based whale optimization algorithm for global optimization. IEEE Access 5:6168–6186
47. Tripathi TA, Sharma K, Bala M (2019) Fake review detection in big data using parallel bbo. Int J Inf Syst Manag Sci 2:2
48. Ashish T, Kapil S, Manju B (2018) Parallel bat algorithm-based clustering using mapreduce. In: Networking communication and data knowledge engineering, Springer, pp 73–82
49. J. Wang, D. Yuan, M. Jiang (2012) Parallel k-pso based on mapreduce. In: 2012 IEEE 14th international conference on communication technology, IEEE, pp 1203–1208
50. Banharnsakun A (2017) A mapreduce-based artificial bee colony for large-scale data clustering. Pattern Recogn Lett 93:78–84
51. Harper FM, Konstan JA (2015) The movielens datasets: history and context. ACM Trans Interactive Intell Syst (tiis) 5(4):1–19
52. Hussain A, Muhammad YS (2019) Trade-off between exploration and exploitation with genetic algorithm using a novel selection operator. Complex Intell Syst 2019:1–14
53. Valian E, Mohanna S, Tavakoli S (2011) Improved cuckoo search algorithm for global optimization. Int J Commun Inf Technol 1(1):31–44
54. Mirjalili S, Gandomi AH, Mirjalili SZ, Saremi S, Faris H, Mirjalili SM (2017) Salp swarm algorithm: a bio-inspired optimizer for engineering design problems. Adv Eng Softw 114:163–191
55. Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. Adv Eng Softw 69:46–61
56. Kennedy J, Eberhart R (1995) Particle swarm optimization. Neural Netw 4:1942–1948
57. Storn R, Price K (1997) Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces. J Global Optim 11:341–359
58. Alabsi F, Naoum R (2012) Comparison of selection methods and crossover operations using steady state genetic based intrusion detection system. J Emerg Trends Comput Inf Sci 3(7):1053–1058