Engineering Applications of Artificial Intelligence 73 (2018) 111–125

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

A combination of objective functions and hybrid Krill herd algorithm for text document clustering analysis

Laith Mohammad Abualigah [a,*], Ahamad Tajudin Khader [a], Essam Said Hanandeh [b]

[a] School of Computer Sciences, Universiti Sains Malaysia, Pulau Pinang, 11800, Malaysia
[b] Department of Computer Information System, Zarqa University, P.O. Box 13132, Zarqa, Jordan

ARTICLE INFO

Keywords:
Combination of objective functions
Hybridization
Krill herd algorithm
K-mean algorithm
Text document clustering

ABSTRACT

Krill herd (KH) algorithm is a novel swarm-based optimization algorithm that imitates krill herding behavior during the searching for foods. It has been successfully used in solving many complex optimization problems. The potency of this algorithm is very high because of its superior performance compared with other optimization algorithms. Hence, the applicability of this algorithm for text document clustering is investigated in this work. Text document clustering refers to the method of clustering an enormous amount of text documents into coherent and dense clusters, where documents in the same cluster are similar. In this paper, a combination of objective functions and hybrid KH algorithm, called, MHKHA, is proposed to solve the text document clustering problem. In this version, the initial solutions of the KH algorithm are inherited from the *k*-mean clustering algorithm and the clustering decision is based on two combined objective functions. Nine text standard datasets collected from the Laboratory of Computational Intelligence are used to evaluate the performance of the proposed algorithms. Five evaluation measures are employed, namely, accuracy, precision, recall, *F*-measure, and convergence behavior. The proposed versions of the KH algorithm are compared with other well-known clustering algorithms and other thirteen published algorithms in the literature. The MHKHA obtained the best results for all evaluation measures and datasets used among all the clustering algorithms tested.

1. Introduction

Web pages and modern applications have become the main sources of enormous amounts of text information owing to the widespread surge of text information over the Internet, such as news sites, learning sites, social media, organizational sites, and digital libraries (Forsati et al., 2008; Fu, 2011; Al-Sai and Abualigah, 2017). To easily manage large numbers of text documents (information), one can use text clustering as an effective unsupervised learning technique for partitioning a set of text documents into coherent and dense clusters (Abualigah et al., 2016b). This technique simplifies text handling for users and involves clusters that each contain relevant documents in terms of intrinsic content (Abualigah et al., 2016a). Text clustering manages a set of unlabeled text documents without any prior knowledge of the document's class label (Karol and Mangat, 2013; Prakash et al., 2014; Abualigah et al., 2018b).

Text document clustering is successfully used to facilitate the process of different systems, such as image recognition, text classification, information retrieval, text categorization, and search engines (Abualigah and Khader, 2017). In this technique, text documents are presented using the vector space model (VSM) to measure the similarity between documents (vector) with cluster centroids. VSM is a common means used to facilitate document representation (Abualigah and Hanandeh, 2015; Singh et al., 2011), where each text document is represented as a vector of terms weights based on the common weighting scheme (i.e., term frequency and inverse document frequency (TF–IDF)) (Kanimozhi and Venkatesan, 2015; Zaw and Mon, 2015).

The *k*-mean is a fast, robust, and highly powerful local search algorithm for solving text document clustering problems (Premalatha and Natarajan, 2010a). This algorithm begins clustering acts with temporary initial cluster centroids. This property is considered a strong advantage of the *k*-mean clustering algorithm over the global clustering algorithm (Abualigah et al., 2016b). Subsequently, the text document is assigned to the most similar cluster centroid on the basis of the maximum similarity between documents and cluster centroids (Jaganathan and Jaiganesh, 2013; Balabantaray et al., 2015).

In recent years, various meta-heuristic optimization algorithms have been proposed to solve difficult optimization problems, such as text document clustering, data mining, image segmentation, computer vision,

* Corresponding author.
E-mail address: laythdyabat@ymail.com (L.M. Abualigah).

https://doi.org/10.1016/j.engappai.2018.05.003
Received 30 June 2017; Received in revised form 27 February 2018; Accepted 9 May 2018

forecasting, information retrieval, and images clustering problems in the medical field (Wang et al., 2015; Nanda and Panda, 2014; Abualigah et al., 2017b; Shehab et al., 2017; Alomari et al., 2017a). The design of meta-heuristic optimization algorithms is based on the pursuit of an optimal solution based on specific roles that attract optimal solutions to solve optimization problems. All the potential values are used to attain various solutions that aid iteratively in reaching an optimal solution, and the terminal value is the optimal solution (Bharti and Singh, 2016a; Alyasseri et al., 2017; Alomari et al., 2017b).

Recently, meta-heuristic algorithms have faced the problem of premature or fast convergence, which diminishes the performance of the global (exploration) search ability. Premature convergence problems can be associated with the quality of initial solutions, such that if the clustering algorithm obtains high-quality solutions, then the global optimum solution is easily achieved (Jaganathan and Jaiganesh, 2013; Premalatha and Natarajan, 2010b; Nanda and Panda, 2014; Abualigah and Hanandeh, 2015; Moayedikia et al., 2015). Several researchers have proposed hybrid strategies that combine local (exploitation) search with global (exploration) search to enhance algorithm diversity during the search process (Wang et al., 2015, 2016, 2014a, b). In the literature on meta-heuristic algorithms, such hybrid strategies have been widely utilized to improve the performance of the krill herd (KH) algorithm by enhancing the original version of the KH algorithm to solve several optimization problems (Bolaji et al., 2016). Moreover, the original KH algorithm does not even guarantee the achievement of a local optimal solution (Premalatha and Natarajan, 2010a; Abualigah et al., 2017).

Artificial bee colony algorithm is applied to select the appropriate text cluster centers for creating text document clusters (Bharti and Singh, 2016a). The authors established two local search models, namely gradient search and chaotic local search in the pure ABC to enhance its exploitation capability. The proposed algorithm is named as chaotic gradient artificial bee colony. The performance of the proposed algorithm is tested using three variant text datasets namely, WebB, Classic 4, and Reuters-21578. The results are compared to global best guided ABC, a variant of the proposed method, memetic ABC, and K-means clustering algorithm. The proposed methods showed encouraging improvements in the quality of clusters and convergence speed.

Four new hybridized bee colony optimization techniques with the k-mean clustering algorithm have been applied (Forsati et al., 2015). These algorithms can avoid the problem of being fixed in the local optima, enhance the global exploitation, and address the shortcoming of this swarm algorithm in local search. An experiment was conducted on data and subset text datasets, and results showed that the proposed algorithm was sufficiently robust compared with the k-mean, genetic algorithm (GA), particle swarm optimization (PSO), and artificial bee colony strategies.

An original harmony search (HS) algorithm is applied to text clustering algorithms to find optimal clusters (Forsati et al., 2013). HS hybridizes with the k-mean algorithm in three ways to acquire an enhanced clustering technique by combining the preliminary efficiency of HS with the strengths of the k-mean algorithm through the latter's local search. Experimental results revealed that the clustering performance of the hybridized HS with k-mean was superior to those of other algorithms.

A new PSO-based cuckoo search (CS) clustering algorithm was proposed to combine the strengths of CS and PSO. The CS solutions were based on PSO solutions (Zaw and Mon, 2015). The new algorithm converts some solutions to successful solutions until an optimal solution is attained. An experiment was conducted on web document datasets and demonstrated that the proposed algorithm performed well in the text clustering area in terms of F-measure values.

A hybridized genetic algorithm with a PSO algorithm was applied to enhance the text clustering technique (Song et al., 2015). In the study, the authors used GA to obtain a global solution as an initial strategy to PSO. Then, a normalized search space was employed to update the PSO positions and obtain a proper range space. Experiments were conducted on subset text datasets, and the hybridization method improved the text clustering method to a greater extent than did the other algorithms.

The KH algorithm is a recent nature-based algorithm inspired by individual krill herding behavior. This algorithm was introduced in 2012 by Gandomi and Alavi to solve function benchmark optimization problems (Gandomi and Alavi, 2012). KH algorithm works to achieve the minimum distance of the krill individuals from the closest food. This algorithm has been successfully utilized to solve many optimization problems, such as numerical optimization, electric and power system problems, text clustering, breast cancer detection, and neural network training (Bolaji et al., 2016; Abualigah et al., 2016b, 2017c).

The contributions of this paper are stated as follows:

(1) Two basic KH algorithm is adapted (KHA) for solving the text document clustering problem; KHA with two genetic operators called KHA1, KHA without the genetic operators called KHA2, and compared with the most common clustering algorithms (k-mean + +, GA, HS, and PSO) to investigate their suitability in solving this problem. The main motivation to solve the text clustering using KHA is that the behavior of the KHA is similar to the behavior of the text clustering process.

(2) Three versions of the hybrid KH algorithm (HKHAs) with k-mean algorithm are proposed; HKHA1 with two genetic operators and HKHA2 with the crossover operator, and HKHA3 with the mutation operator. The proposed hybrid strategy also involves developing an efficient clustering algorithm to produce results that are less dependent on selecting initial cluster centroids and accurate document clusters. The HKHAs with k-mean mainly aim to improve the local (exploitation) search and global (exploration) search capabilities to obtain the optimal value of the objective function. HKHAs seek to tackle trapped in the local optimum, premature convergence and increase convergence speed of the basic KH algorithm.

(3) A combination of objective functions is proposed for the best local search concept in k-mean clustering algorithm, called MHKHA, to improve the performance of the hybrid KH algorithm by making an accurate decision during the assigning each document into the proper cluster centroid.

To demonstrate the performance and speed of the proposed algorithms, experiments were conducted using nine common text benchmark datasets commonly used in the domain of the text clustering. Results reveal that the MKHA is sufficiently robust and obtains the best results in almost all datasets compared with other comparative clustering algorithms, including other KH versions. Generally, the performance of the KH algorithm is not as good as that of MKHA in terms of accuracy, precision, recall, and F-measure.

The remainder of this paper is organized as follows. Section 2 presents the preliminaries to the text document representation. Section 3 explains the k-mean text clustering algorithm. Section 4 discusses the basic KH algorithm. Section 5 introduces the modeling of the KH algorithm for the text document clustering. Section 6 introduces the proposed KH algorithms for text document clustering. Section 7 illustrates the experiments and results. Finally, Section 8 provides the conclusion and future work.

2. Preliminaries

This section discusses the modeling of the text document clustering as an optimization problem.

2.1. Descriptions and formulations of text document clustering problem

A collection of text documents D is split into K clusters, where D is represented as a vector of text documents in Eq. (1) as follows:

$$D = d_1, d_2, \ldots, d_i, \ldots, d_n \tag{1}$$

In Eq. (1), d_i is the document number i in collection D, and n is the number of all text documents in D (Premalatha and Natarajan, 2010a; Abualigah et al., 2016a, 2018b). Moreover, each document is represented as a vector $d_1 = w_{11}, w_{12}, \ldots, w_{1j}, \ldots, w_{1t}$, where d_1 is the

document number one with length of *t*, and w_{ij} represents the weight of term *j* in document number *i*, calculated by Eq. (2):

$$w_{ij} = TFIDF(i,j) = tf(i,j) * (\log \frac{n}{df(j)}), \quad (2)$$

where $tf(i,j)$ is the frequency of term *j* in document *i*, *n* is the number of all text documents in *D*, and *df (j)* is the number of documents that contain term number *j* (Premalatha and Natarajan, 2010a; Mohammed et al., 2015; Abualigah et al., 2016b).

The clustering algorithm handles cluster *k* by its own cluster centroid c_k, which is represented as a vector of term weight $C_k = c_1, c_2, \ldots, c_j, \ldots, c_t$, where c_k is the *k*th cluster centroid, c_1 is the value of the position 1 of the cluster centroid number *k*, and *t* is the length of clusters centroids. Notably, cosine similarity measure is used by the clustering algorithm to calculate the similarity score between each document with all clusters centroids (Abualigah et al., 2016b,a).

2.2. Similarity measure

Text document clustering attempts to group similar text documents into the same cluster, where dissimilar documents are grouped in different clusters. The cosine similarity measure is used to compute the similarity score between each document with clusters centroids by Eq. (3) (Zaw and Mon, 2015; Abualigah et al., 2016b; Singh et al., 2011).

$$Cosine(d_1, c_3) = \frac{d_1 * c_3}{||d_1|| * ||c_3||}, \quad (3)$$

where $Cosine(d_1, c_3)$ is the similarity between the document number 1 (d_1) and the cluster centroid number 3 (c_3). As aforementioned, d_1 is a vector of terms weight of length *t*, $\|d_1\|$ is the summation of the vector 1, and $\|c_3\|$ is the summation of vector 3 (centroid of cluster number 3). This measure provides a value close to one if document number 1 is similar to cluster centroid number 3, and zero if document number 1 is dissimilar to cluster centroid number 3 (Karol and Mangat, 2013; Prakash et al., 2014; Abualigah et al., 2016a).

2.3. Distance measure

Euclidean distance is a standard measure used in the domain of the text clustering to compute the distance (dissimilarity) between the documents and their clusters centroids as shown in Eq. (4). Normally, distance values are between (0, 1), although it is unlike the values of the cosine similarity measure. Where, if the distance value between a document and a cluster centroid was close to 0, it means that this document was very close (similar) to the cluster centroid. If the distance value close to 1, it means that this document was not close (dissimilar) to the cluster centroid (Mohammed et al., 2015).

$$Dis_{(d4,c2)} = (\sum_{j=1}^{t} \left| w_{d,j} - w_{c,j} \right|^2)^{1/2}, \quad (4)$$

Eq. (4) presents the distance between the document number 4 and the cluster centroid number 2. Where, $w_{d,j}$ is the weight of term *j* in the document number 4, and $w_{c,j}$ is the weight of term *j* in the cluster centroid number 2.

2.4. The proposed combination of objective functions

In the MHKHA, a new formula has been proposed to improve the performance of the hybrid KH algorithm, which combines two different measures as objective function to ensure that the decision making during the clustering process is accurate. This combination consists of two different measures, which are commonly used in the text clustering domain separately as mentioned previously, namely, cosine similarity measure (see Eq. (3)) and Euclidean distance (see Eq. (4)) measure. Accurate similarity and distance measures from several similarity and distance measures in the literature were used to determine the advantages of both measures (Rao et al., 2017; Abualigah and Khader, 2017). Eq. (5) represents the equation of the proposed combination of objective functions function (Multi-obj):

$$Multi - obj = Cos(d_1, c_2) * (1 - Dis(d_1, c_2)) \quad (5)$$

Where, $Cos(d_1, c_2)$ is the value of the cosine similarity between document 1 and the centroid of cluster 2 calculated using Eq. (3) and $Dis(d_1, c_2)$ is the value of the Euclidean distance between document 1 and the centroid of cluster 2 calculated using Eq. (4).

3. *K*-mean clustering algorithm

The *k*-mean text clustering is an efficient and robust algorithm utilized in the domain of document clustering. The main components of this algorithm are the number of clusters *K*, initial cluster centroids, partitioning of each document to similar centroids, and the similarity score between the document and the centroid of each cluster; these cluster centers then update every iteration and the termination criteria (Forsati et al., 2013; Tunali et al., 2016; Prakash et al., 2014).

A strong advantage of the *k*-mean clustering algorithm is the process of choosing the initial cluster centroids (Abualigah et al., 2017a). The algorithm partitions a collection of text documents *D* with a high-dimensional space into a subset of coherent clusters K. The *k*-mean clustering algorithm uses the similarity score to assign each document for the similar centroid using Eq. (3). This aspect uses data matrix $A_{(n*K)}$, where *n* is the number of all text documents and *K* is the number of clusters. Each document is represented as a vector of term weight through Eq. (1). Meanwhile, t is the number of unique text features (terms), and the *k*-mean algorithm searches the optimal *n*K* matrix (Abualigah et al., 2016a; Abualigah and Khader, 2017). This procedure is reviewed in Algorithm 1.

Algorithm 1: K-mean clustering algorithm [22]

1: Input: *D* is a collection of text documents, *K* is the number of clusters.
2: Output: Assign *D* to *K*.

3: (Termination criteria)
4: Randomly choose *K* documents as clusters centroids.
5: Initialize matrix $A_{(n*k)}$ as zeros
6: For all *d* in *D* do
7: Let *j* = argmax kε{1….K}, based on *Multi-obj* (*di*, *ck*).
8: Assign d_i to the cluster number *j*, *A*[*i*][*j*] = 1.
9: End for
10: Update the clusters centroids

The *k*-mean algorithm operates to find the local optimum solution because the algorithm is affected by the initial cluster centers during the clustering process. This effect is achieved because the algorithm reflects the quality of document clusters. Results are enhanced regularly if the initial cluster centers are selected close to the main intrinsic clusters. Hence, the quality of document clusters depends on the initial centroids. If the text documents in a given collection hold similar characteristics, the *k*-mean function thoroughly in improving the clusters centroids to find optimal clusters (Forsati et al., 2013; Singh et al., 2011; Abualigah et al., 2016a).

4. Basic krill herd algorithm

The KH algorithm is a recent meta-heuristic algorithm that has been successfully used to solve global optimization problems by simulating the herding behavior of krill individuals (Gandomi and Alavi, 2012; Bolaji et al., 2016; Abualigah et al., 2017). Krill positions are updated using three effect motions (i.e., movement induced by other krill individuals, foraging activity, and random diffusion). Sensing distance refers to the

objective function of the KH algorithm. This component is used to find the closest food by computing the distance score of each krill individual with food and comparing against the highest density (Abualigah et al., 2016b, 2017c).

4.1. Position update

The position of the ith krill individual under the interval I is updated to the interval $I+\Delta I$ by using Eq. (6) (Gandomi and Alavi, 2012; Bolaji et al., 2016; Abualigah et al., 2016b):

$$x_i(I+\Delta I) = x_i(I) + \Delta I \frac{dX_i}{d_s} \tag{6}$$

where $x_i(I+\Delta I)$ represents the value of the next krill individual position, and $x_i(I)$ represents the current position of solution number i at iteration I. Notably, ΔI is considered the essential constant and should be tuned carefully based on the text clustering optimization problem (Bolaji et al., 2016; Abualigah et al., 2016b). The decision value for the ith krill individual ($\frac{dx_i}{ds}$) is updated using Eq. (7) (Gandomi and Alavi, 2012):

$$\frac{dx_i}{ds} = F_i + N_i + D_i \tag{7}$$

where F_i is the foraging motion factor, N_i is the motion affected by other krill individuals, and D_i is the physical diffusion of the ith position.

4.1.1. Motion affected by other krill individuals

The motion of each krill individual can be calculated by Eq. (8). The direction of the first movementa_i is obtained using Eq. (9) on the basis of some factor that includes the objective effect, local effect, and repulsive head density (Abualigah et al., 2016b, 2017). This parameter is obtained through the two main equations, Eqs. (10) and (13):

$$N_i^{new} = N^{max} a_i + a_i^{target} \tag{8}$$

where

$$a_i = a_i^{local} + a_i^{target} \tag{9}$$

N^{max} is the parameter used to adjust the maximum affected speed after the KH algorithm parameter is tuned by taking the value of the maximum induced speed (0.01) (Abualigah et al., 2016b). ωn is the inertia weight for the effect motion between (0, 1) based on individual similarity. N^{old} is the last value introduced by other krill individuals, a_i^{local} is the local effect, and a_i^{target} is the target effect (Abualigah et al., 2016b). The effect of krill neighbors on individual movement is calculated using Eq. (10):

$$a_i^{local} = \sum_{j=1}^{NN} K'_{i,j} * x'_{i,j} \tag{10}$$

where

$$X'_{i,j} = \frac{x_j - x_i}{\left\| x_j - x_i \right\| + \varepsilon} \tag{11}$$

$$K'_{i,j} = \frac{K_i - K_j}{K^{worst} - K^{best}} \tag{12}$$

K^{worst} and K^{best} represent the worst and best objective function values of the krill individual at a particular point, respectively. K_i denotes the objective function for the individual i, K_j represents the objective function for the jth neighbor, x represents the related individual, ε is the small positive number (0.09), x_j is the current position, and x_j is the jth neighbor. The parameter is in the range 1 to NN, where NN represents the number of krill individual positions, which is equal to the total number of documents n (Gandomi and Alavi, 2012; Bolaji et al., 2016; Abualigah et al., 2016b).

$$a_i^{target} = C^{best} K'_{i,best} x'_{i,best} \tag{13}$$

where

$$C^{best} = 2(rand + \frac{I}{I_{max}}) \tag{14}$$

C^{best} represents a coefficient score of the krill individual, a_i^{target} represents a coefficient value that helps reach the global optimum, *rand* is the rand number (0, 1), I represents the current KH iteration, and Imax is the max number of KH iterations (Abualigah et al., 2016b).

4.1.2. Foraging motion

Foraging motion operates on the basis of two main parts: current food location and previous food location (Gandomi and Alavi, 2012; Abualigah et al., 2016b). The foraging motion of the ith krill individual defines by Eq. (15) as follows:

$$F_i = V_f B_i + \omega_f F_i^{old} \tag{15}$$

where

$$B_i = B_i^{food} + B_i^{best} \tag{16}$$

V_f represents the parameter of foraging speed, and the value of the foraging speed is obtained (0.03) in Abualigah et al. (2016b). ω_f is the inertia weight value taken between (0, 1), F_i^{old} represents the current foraging speed, B_i^{food} represents food attractiveness, and B_i^{best} represents the best affected krill individual (Bolaji et al., 2016; Abualigah et al., 2017).

4.1.3. Physical diffusion

This factor in the KH algorithm is considered a stochastic means to find a maximum diffusion speed for each vector direction. The physical diffusion of the ith krill individual is calculated using Eq. (17) (Bolaji et al., 2016; Abualigah et al., 2016b).

$$D_i = D^{max}\left(1 - \frac{I}{I_{max}}\right)\delta \tag{17}$$

where D^{max} represents the parameter of the maximum diffusion speed, which is obtained as (0.008) in Abualigah et al. (2016b), and δ is a random vector by values between (−1, 1) (Gandomi and Alavi, 2012; Abualigah et al., 2016b).

4.1.4. Genetic operators of KH algorithm

Genetic operators are used to improve the performance of the KH algorithm. These operators are taken from the classical evolutionary algorithms (Premalatha and Natarajan, 2010b; Abualigah et al., 2016b).

4.1.5. Crossover of KH algorithm

Crossover is the master operator in the genetic algorithm. This operator functions by swapping positions between components of the selected two solutions to reach a global optimum solution (Abualigah et al., 2016b). The crossover operator is controlled by the crossover probability (Cr), which is defined as $Cr = (0.2^* K_{i,best})$. The jth position of the ith krill individual is updated using Eq. (18) (Gandomi and Alavi, 2012; Bolaji et al., 2016):

$$x_{ij} = \begin{cases} x_{rj} & if\ rand < C_r \\ & where\ r = 1, 2, \ldots, NN; r \neq i \\ x_{ij} & otherwise \end{cases} \tag{18}$$

4.1.6. Mutation operator

Mutation is a highly valuable operator of evolutionary algorithms that works by flipping some positions of the selected solutions to obtain a global optimum solution (Abualigah et al., 2016b). The mutation operator is controlled by the mutation probability (Mu), which is defined as $Mu = 0.05/K_{i,best}$. The jth position of the ith krill individual may be updated using Eq. (19) (Gandomi and Alavi, 2012; Bolaji et al., 2016) as follows:

$$x_{ij} = \begin{cases} x_{rj} & if\ rand < Mu \\ x_{ij} & otherwise \end{cases} \tag{19}$$

Fig. 1. Solution representation.

5. Modeling krill herd for text document clustering problem

This section describes the modeling of text document clustering using the KH algorithm to find optimal clusters. The following subsections describe KH acts, which are used to perform the clustering technique.

5.1. Krill herd solution representation

KH algorithm performs text document clustering through a set of solutions S (herd). Each solution is represented as a vector of length n, where n is the number of all documents, and each document signifies a krill individual in the acts of the KH algorithm. Each decision variable (krill individual) belongs to one cluster centroid $[1 \ldots K]$. Each solution also contains a set of K centroids $C = c_1, c_2, \ldots, c_k, \ldots, c_K$, $C = (c_1, c_2, \ldots, c_i, \ldots, c_k)$ where $c_k c_1$ represents the kth cluster centroid (Abualigah et al., 2016a). Fig. 1 shows an example of one solution representation. This solution shows eight documents divided into three clusters, where d_1 belongs to cluster label 3 and d_5 belongs to cluster label 2. For example, class 3 contains documents by the numbers 1, 2, and 7.

5.2. Krill herd memory (KHM)

The KHM is initialized by random values depending on the search space intervals between $[1 \ldots K]$. Each vector represents one solution in the KHM that corresponds to a number of cluster centroids K. The size of KHM is based on some solutions and a number of documents (S*n). In the KH algorithm, uniform distribution is not important in filling the KHM because this distribution is not sensitive to solution initialization (Abualigah et al., 2016b). The KHM is expressed by Eq. (20).

$$KHM = \begin{bmatrix} x_{1,1} & x_{1,2} & x_{1,j} & \ldots & \ldots & x_{1,n} \\ x_{2,1} & x_{2,2} & x_{2,i} & \ldots & \ldots & x_{2,n} \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ x_{S,1} & x_{S,2} & x_{S,i} & \ldots & \ldots & x_{S,n} \end{bmatrix} \quad (20)$$

5.3. Updating the cluster centroid

The cluster centroid is the main important step in text document clustering because all clustering steps are performed based on the cluster centroid (Karol and Mangat, 2013; Tunali et al., 2016). This cluster centroid is used to decide if the document should belong to the cluster. The cluster centroid of cluster c_j is then computed by Eq. (21) as follows:

$$c_j = \frac{1}{n_i} \sum_{\forall d_i \in c_i} d_i \quad (21)$$

where d_i denotes that document i belongs to the c_j centroid of cluster j, and n_i represents all the number of documents that belong to cluster c_i.

5.4. Fitness function

Fitness function is used to evaluate KH solutions on the basis of cluster quality. The average similarity document centroid (ASDC) is the fitness function of this work and is calculated by Eq. (22). This function is considered an external measure, and its value is based on the calculation of the internal measure (cosine similarity) between each cluster's documents with its own centroid. As aforementioned, combination of objective functions function is used to compute the similarity of each document with all clusters centroids in the k-mean clustering algorithm. However, the KH algorithm uses the similarity to evaluate each cluster components depending on its own documents by Eq. (22) (Forsati et al., 2013; Zaw and Mon, 2015; Abualigah et al., 2016b):

$$ASDC = \frac{\sum_{i=1}^{k} \left\{ \frac{\sum_{j=1}^{n_i} Multi\text{-}obj(c_i, d_{ij})}{n_i} \right\}}{K} \quad (22)$$

where n_i represents all the numbers of documents in cluster i, $Multi - obj\,(c_i, d_{ij})$ is the similarity value between the jth document with the ith cluster centroids, and d_{ij} represents the jth document of cluster i.

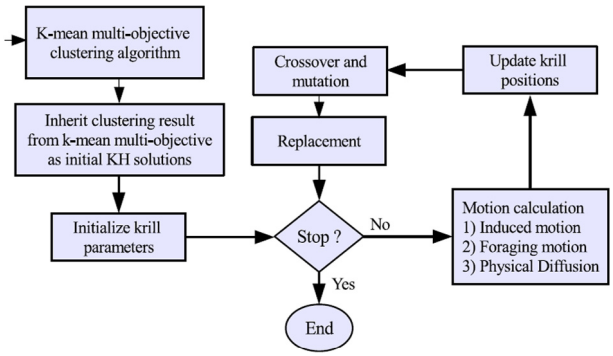

Fig. 2. The sequence of the MHKHA procedures.

6. Proposed MHKHA for text clustering

In this section, we present a combination of objective functions and hybrid KH algorithm with the k-mean algorithm for text document clustering as shown in Fig. 2. This algorithm adopts the combination of objective functions in the k-mean clustering algorithm to fill initial solutions of the KHM for enhancing the performance of the KH algorithm. This feature worthy to avoid the trapping in the local (exploitation) search and prevent the premature convergence in some cases. The MHKHA connects the explorative power of the KH algorithm to the speed of the k-mean algorithm in solution configuration along with the exploitative power.

Algorithm 2 illustrates the proposed MHKHA, and the algorithm covers two stages. The first stage is the combination of objective functions for the k-mean clustering algorithm (Algorithm 1, lines 3–12). This stage seeks to find the local optimum within a short period (50 iterations) to avoid consuming high computation time. These solutions are obtained as the initial solutions of the KHM to avoid trapping in a local optimum. The second stage is the KH clustering algorithm (Algorithm 1, lines 16–40), which discovers the attainment of the global optimum solution by inheriting KH solutions from the combination of objective functions of the k-mean algorithm. This stage seeks to improve the search ability of the KH algorithm and tackles the trapping in the local optimum by a long period (1000 iterations). Finally, finding the balance between local (exploitation) and global (exploration) search capabilities is necessary. In this algorithm, the explorative power of MHKHA derived from the k-mean algorithm and the power of the KH algorithm is encouraged in every iteration by obtaining a high-quality solution to achieve accurate clusters.

Algorithm 2: MHKHA for text clustering.

1: Initialization of k-mean parameters K, and $KImax$.
2: Initialization of KH parameters: $MaxI$, S, and others.
3: For l = 1 to S do \\Note, k-mean starting, l is the number of first solution in the KHM solution.
4: Randomly select K documents as the initial cluster centroid
5: For KI = 1 to $KImax$ do \\ for k-mean algorithm
6: Initialize matrix A as zeros
7: For j = 1 to D do
8: Let j = argmax $k\varepsilon\{1\ldots K\}$, on the basis of $Multi\text{-}obj$ (di,ck)
9: Assign di to the cluster j, i.e., $A[i][j]=1$
10: Update the clusters centroids
11: Endfor
12: Endfor
13: Convert matrix A as a matric of solutions (KHM).
14: $S(l) = A$, note that each k-mean generation is one solution for the KH memory.
15: Endfor
16: Initialization of KHM using S, which is the k-mean results. \\Note KH Starting
17: For i =1 to S do
18: For j = 1 to n do
19: Compute the clusters centroids
20: Compute fitness function of each krill by using $ASDC$
21: Endfor
22: Endfor
23: Sort the krills and find $x$best , where best from [1, 2, ..., S]
24: While $I \neq Imax$ do
25: For i = 1 to S do
26: Perform the three motion calculations
27: $xi(I+dI) = xi(I)+\Delta I\ (dxi/ds)$
28: Compute the clusters centroids
29: Evaluate each krill using $ASDC$
30: Endfor
31: For i = 1 to S do
32: Apply KH operators to the KH memory.
33: Crossover
34: Mutation
35: Endfor
36: Replace the worst krill with the best krill
37: Sort the krills and find $xbest$
38: $I = I$+1
39: Endwhile
40: Return xbest

7. Experiments and results

In this section, we compared the proposed methods (i.e., KHA1, KHA2, HKHA1, HKHA2, HKHA3, and MHKHA) with other comparative algorithms in the text clustering domain. The comparisons were performed based on the quality of the clusters and the convergence speed by different text document datasets. We developed the MHKHA for the document clustering application using Matlab (7.10.0) software with different CPU and RAM 4G.

7.1. Evaluation measures

Cluster quality is measured in the text clustering domain by accuracy, precision, recall, and F-measure (Abualigah et al., 2018). These common evaluation measures are used to evaluate document clusters and measure the true partition in each cluster in accordance with the class labels for each cluster document. The proposed algorithm is also evaluated through these evaluation criteria (Karol and Mangat, 2013; Kanimozhi and Venkatesan, 2015; Bharti and Singh, 2016a; Abualigah et al., 2016a). The following subsections show the evaluation measures adopted in the present study.

Table 1
Text document datasets.

Dataset	No. of documents	No. of terms	No. of clusters
DS1	299	1725	4
DS2	333	4339	4
DS3	204	5832	6
DS4	313	5804	8
DS5	414	6429	9
DS6	878	7454	10
DS7	913	3100	10
DS8	2000	7748	4
DS9	18828	45433	20

7.1.1. Precision measure

Precision (P) measure is the ratio of all true relevant documents to the total number of documents in all clusters. This measure is computed for each cluster on the basis of the given true class label by Eq. (23) (Kanimozhi and Venkatesan, 2015; Abualigah et al., 2016b).

$$P(i,j) = \frac{n_{ij}}{n_j} \tag{23}$$

where $P(i, j)$ is the precision value of class i in cluster j, n_{ij} is the number of true members of class i in cluster j, and n_j is the number of all members of the cluster label j.

7.1.2. Recall measure

Recall (R) measure is the ratio of the true number of relevant documents to that of all the clusters' documents. The recall is computed for each cluster on the basis of the given class label using Eq. (24) (Abualigah et al., 2016a) as follows:

$$R(i,j) = \frac{n_{ij}}{n_i} \tag{24}$$

where $R(i, j)$ is the recall value of class i in cluster j, and n_i is the number of true members of class i.

7.1.3. F-measure

F-measure is a popular evaluation measure used in the text clustering domain. This parameter is based on the gathering precision and recall measures. A perfect document clustering technique leads to an F-measure value close to 1. The F-measure for the class i in cluster j is calculated using Eq. (25) (Karol and Mangat, 2013; Kanimozhi and Venkatesan, 2015; Abualigah et al., 2016b,a):

$$F(i,j) = \frac{2 * P(i,j) * R(i,j)}{P(i,j) + R(i,j)} \tag{25}$$

where the F-measure for all clusters is calculated using Eq. (26):

$$F = \sum_j \frac{n_j}{n} \max_i F(i,j) \tag{26}$$

where n is the number of all documents in collection D.

7.1.4. Accuracy measure

Cluster accuracy is a common evaluation measure used to compute the percentage of assigned true text documents to each cluster. This parameter is calculated by Eq. (24) (Karol and Mangat, 2013; Abualigah et al., 2017a):

$$Ac = \sum_{i=1}^{K} \frac{1}{n} P(i,j) \tag{27}$$

where K is the number of total clusters and $P(i, j)$ is the precision value of class i in cluster j.

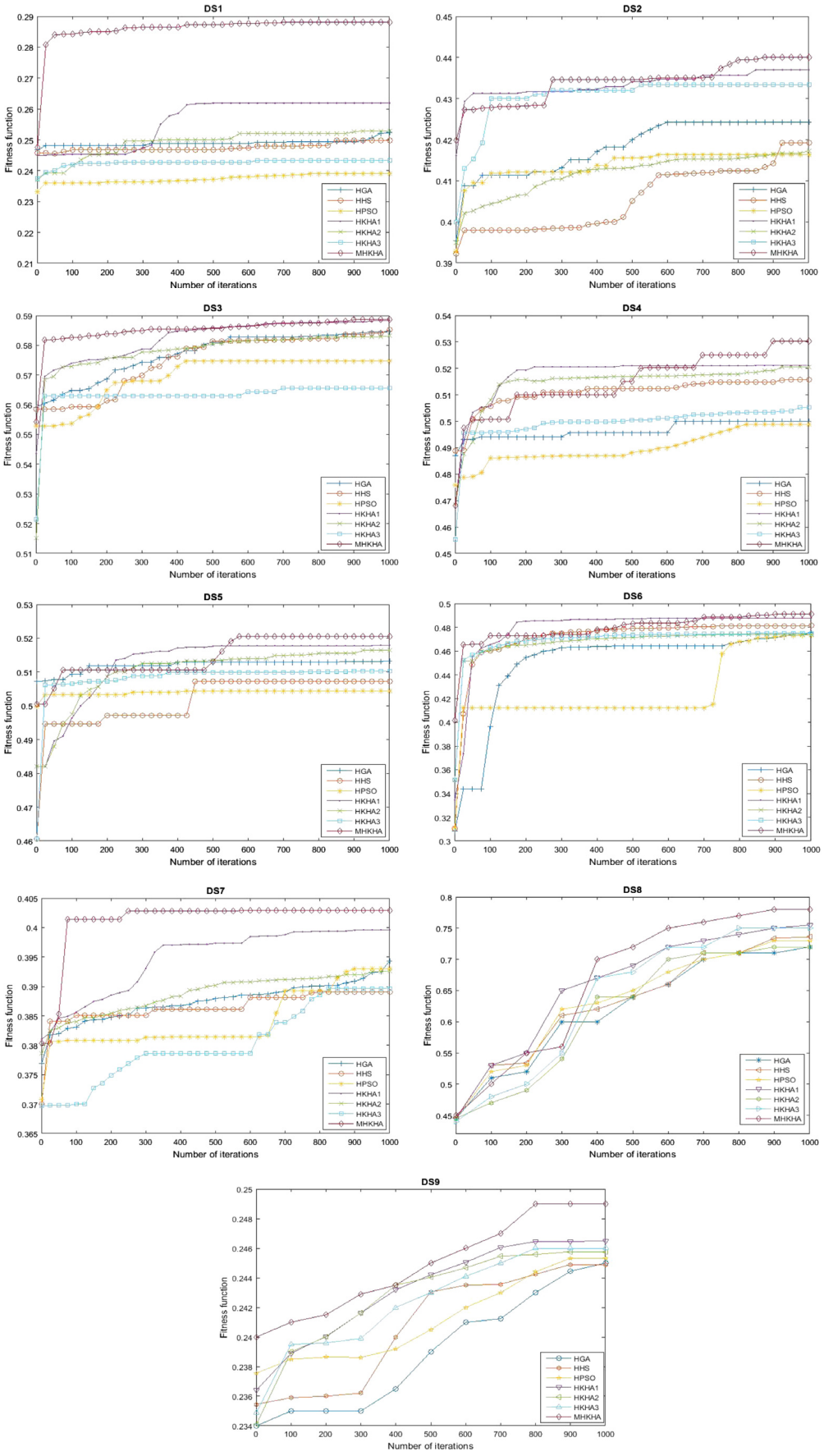

Fig. 3. Convergence behavior of the proposed clustering algorithms using fitness function.

Table 2
Characteristics of the KH versions.

No.	Name	Discretion	Crossover	Mutation
1	KHA1	Basic KH algorithm	Yes	Yes
2	KHA2	Basic KH algorithm	No	No
3	HKHA1	Hybrid KH with k-mean algorithm	Yes	Yes
4	HKHA2	Hybrid KH with k-mean algorithm	Yes	No
5	HKHA3	Hybrid KH with k-mean algorithm	No	Yes
6	MHKHA	Combination of objective functions and hybrid KH algorithm	Yes	Yes

Table 3
Convergence scenarios of the basic krill herd algorithm (BKHA).

Scenario	S	V_f	D_{max}	N_{max}	The obtained best results
Standard Gandomi and Alavi (2012)	–	0.020	0.002	0.010	
1	10	0.020	0.002	0.010	2
2	20	0.020	0.002	0.010	**24**
3	30	0.020	0.002	0.010	3
4	40	0.020	0.002	0.010	4
5	50	0.020	0.002	0.010	3
6	20	0.005	0.002	0.010	2
7	20	0.010	0.002	0.010	4
8	20	0.030	0.002	0.010	**19**
9	20	0.040	0.002	0.010	1
10	20	0.070	0.002	0.010	10
11	20	0.030	0.008	0.010	5
12	20	0.030	0.030	0.010	4
13	20	0.030	0.040	0.010	0
14	20	0.030	0.080	0.010	**25**
15	20	0.030	0.100	0.010	2
16	20	0.030	0.008	0.005	5
17	20	0.030	0.008	0.020	0
18	20	0.030	0.008	0.030	3
19	20	0.030	0.008	0.050	1
20	20	0.030	0.008	0.100	**27**
Best Scenario (20)	**20**	**0.030**	**0.008**	**0.100**	**27**

7.2. Text clustering datasets

To fairly compare the effectiveness of the proposed algorithms, we used nine text datasets with different characteristics, including number of clusters, documents, and terms. Text clustering standard benchmark datasets are available at http://sites.labic.icmc.usp.br/text_collections/ by numerical form after term extraction.

Table 1 shows the seven datasets used in this paper. The first dataset (DS1), derived from the CSTR, contains 299 documents on technical reports belonging to four clusters. The second dataset (DS2), derived from the SyskillWebert, contains 333 documents on the Web pages belonging to four clusters. The third dataset (DS3), derived from the Trace, contains 204 documents on tr32 belonging to six clusters. The fourth dataset (DS4), derived from the Trace, contains 313 documents on tr12 belonging to nine clusters. The fifth dataset (DS5), derived from the Trace, contains 414 documents on tr11 belonging to nine clusters. The sixth dataset (DS6), derived from the Trace, contains 878 documents on tr41 belonging to 10 clusters. The seventh dataset (DS7), derived from the OHSUMED, contains 913 documents on MIDLINE belonging to 10 clusters. The eighth dataset (DS8), derived from classic4, contains 2000 documents on MIDLINE belonging to 4 clusters (CACM, CRAN, CISI, and MED), 500 documents from each category. Finally, the ninth dataset (DS9), derived from the 20NEWSGRUP, contains 18828 documents on news belonging to 20 clusters.

7.3. Experimental setup

We tested two versions of the basic KH algorithm (e.g., KHA1 and KHA2), three versions of the hybrid KH algorithm (e.g., HKHA1, HKHA2, and HKHA3), and a version of combination of objective functions with hybrid KH algorithm (MHKHA) to solve the text document clustering problem. Table 2 shows the characteristics of the KH versions. These versions were investigated to determine the more efficient and effective version that would obtain accurate clusters with high convergence rate. All versions of the KH algorithm were compared with the other well-known hybrid versions (i.e., harmony search, genetic algorithm, and particle swarm optimization, namely, HHS, HGS, and HPSO) and the other published clustering methods in the literature.

The results shown in this paper are based on the average of over 20 algorithm runs. The clustering algorithms administered 1000 iterations in each run because 1000 iterations are sufficient for the convergence of the global search algorithms. Meanwhile, the k-mean administered 50 iterations in each run because 50 iterations are sufficient for the convergence of the local search algorithms (Forsati et al., 2015; Abualigah et al., 2016b).

Table 4
The results of BKHA convergence scenarios (Scenario (1) through Scenario (5)).

Dataset	Measure	Scenario number				
		1	2	3	4	5
DS1	Accuracy	0.4511	**0.4803**	0.4730	0.4548	0.4782
	Precision	0.4444	**0.4903**	0.4821	0.4647	0.4888
	Recall	0.4026	0.4442	0.4218	0.4239	**0.4486**
	F-measure	0.4217	**0.4519**	0.4477	0.4422	0.4169
DS2	Accuracy	0.6252	**0.6363**	0.5819	0.5756	0.5948
	Precision	0.5164	**0.6692**	0.6409	0.6375	0.6284
	Recall	0.6260	**0.6410**	0.5726	0.5656	0.5845
	F-measure	0.6491	**0.6545**	0.6041	0.5987	0.6052
DS3	Accuracy	0.4151	**0.4188**	0.4193	0.4123	0.4020
	Precision	0.3498	0.3562	0.3438	**0.3596**	0.3456
	Recall	0.3465	**0.3535**	0.3516	0.3302	0.3402
	F-measure	0.3474	**0.3584**	0.3465	0.3475	0.3421
DS4	Accuracy	0.5271	**0.5304**	0.5047	0.5074	0.5022
	Precision	0.5505	**0.5559**	0.5410	0.5401	0.5422
	Recall	0.5120	**0.5126**	0.4892	0.4756	0.4851
	F-measure	0.5334	**0.5360**	0.5135	0.5135	0.5115
DS5	Accuracy	0.4783	0.4745	0.4461	**0.4787**	0.4471
	Precision	**0.4168**	0.4143	0.3970	0.3921	0.3962
	Recall	0.3957	0.3950	0.3711	**0.4129**	0.3721
	F-measure	0.4052	0.4033	0.3832	**0.4074**	0.3852
DS6	Accuracy	**0.4342**	0.4167	0.4266	0.4301	0.4258
	Precision	0.4011	**0.4235**	0.3917	0.3998	0.3820
	Recall	0.4071	0.4069	**0.4138**	0.4116	0.3910
	F-measure	0.4038	**0.4100**	0.4047	0.4003	0.3898
DS7	Accuracy	0.3490	**0.3586**	0.3558	0.3558	0.3544
	Precision	0.3520	0.3550	0.3610	0.3610	**0.3617**
	Recall	0.3213	**0.3215**	0.3198	0.3198	0.3196
	F-measure	0.3358	0.3371	0.3391	0.3391	**0.3392**
DS8	Accuracy	0.7254	**0.7290**	**0.7295**	0.7154	0.7231
	Precision	0.7651	0.7720	**0.7723**	0.7698	0.7698
	Recall	0.7328	**0.7399**	0.7259	0.7289	**0.7287**
	F-measure	0.7612	**0.7655**	0.7548	0.7588	**0.7610**
DS9	Accuracy	0.3711	**0.3741**	0.3712	0.3659	0.3695
	Precision	0.3741	**0.3799**	0.3749	0.3641	0.3694
	Recall	0.3841	**0.3888**	0.3789	0.3812	0.3844
	F-measure	0.3809	**0.3855**	0.3715	0.3745	0.3719
Summation		2	24	3	4	3

Table 5
The results of BKHA convergence scenarios (Scenario (6) through Scenario (10))

Dataset	Measure	Scenario number				
		6	7	8	9	10
DS1	Accuracy	0.4491	**0.4848**	0.4545	0.4640	0.4770
	Precision	0.4356	**0.4758**	0.4482	0.4561	0.4737
	Recall	0.3964	0.4413	0.4017	0.4094	**0.4461**
	F-measure	0.4142	0.4570	0.4226	0.4308	**0.4584**
DS2	Accuracy	0.7548	0.7630	**0.7778**	0.7548	0.7417
	Precision	0.7709	0.7743	**0.7893**	0.7709	0.7571
	Recall	0.7653	0.7628	0.7799	0.7653	**0.7867**
	F-measure	0.7681	0.7684	**0.7743**	0.7681	0.7716
DS3	Accuracy	**0.4196**	0.4068	0.3995	0.4164	0.4154
	Precision	0.3311	0.3348	0.3284	0.3326	**0.3462**
	Recall	0.3525	0.3423	0.3395	**0.3585**	0.3491
	F-measure	0.3459	0.3324	0.3320	0.3345	**0.3466**
DS4	Accuracy	**0.5330**	0.5119	0.5202	0.5210	0.5322
	Precision	0.5540	0.5398	0.5533	0.5476	**0.5553**
	Recall	0.5238	0.4943	0.5051	0.5067	**0.5170**
	F-measure	0.5429	0.5159	0.5277	0.5261	**0.5353**
DS5	Accuracy	0.4661	0.4487	**0.4794**	0.4544	0.4629
	Precision	0.4041	0.3885	**0.4114**	0.4089	0.3993
	Recall	0.3911	0.3663	**0.4065**	0.3849	0.3793
	F-measure	0.3967	0.3760	**0.4086**	0.3945	0.3884
DS6	Accuracy	0.4257	0.4567	**0.4578**	0.4474	0.4505
	Precision	0.3963	0.4090	**0.4135**	0.4159	0.4206
	Recall	0.4081	0.4289	**0.4188**	0.3954	0.4094
	F-measure	0.4013	0.4111	**0.4165**	0.4001	0.4133
DS7	Accuracy	0.3490	**0.3586**	0.3558	0.3558	0.3544
	Precision	0.3520	0.3550	0.3610	0.3610	**0.3617**
	Recall	0.3213	**0.3215**	0.3198	0.3198	0.3196
	F-measure	0.3358	0.3371	0.3391	0.3391	**0.3392**
DS8	Accuracy	0.7245	0.7154	**0.7399**	0.7115	0.7236
	Precision	0.7720	0.7624	**0.7800**	0.7600	0.7645
	Recall	0.7345	0.722	**0.7399**	0.7299	**0.**7290
	F-measure	0.7700	0.7645	**0.7702**	0.7588	**0.**7610
DS9	Accuracy	0.3844	0.3816	**0.3845**	0.3785	0.3826
	Precision	0.3714	0.3754	**0.3811**	0.3800	0.3778
	Recall	0.3856	0.3868	**0.3921**	0.3911	0.3912
	F-measure	0.3819	0.3817	**0.3844**	0.3819	0.3798
Summation		2	4	19	1	10

Table 6
The results of BKHA convergence scenarios (Scenario (11) through Scenario (15)).

Dataset	Measure	Scenario number				
		11	12	13	14	15
DS1	Accuracy	0.4297	0.4374	0.4687	**0.4775**	0.4672
	Precision	0.4047	0.4267	0.4603	**0.4619**	0.4574
	Recall	0.3831	0.3885	0.4367	**0.4379**	0.4233
	F-measure	0.3929	0.4060	0.4409	**0.4412**	0.4381
DS2	Accuracy	0.7535	0.7548	0.7455	**0.7567**	0.7564
	Precision	0.7589	**0.7709**	0.7658	0.7273	0.7648
	Recall	0.7503	**0.7653**	0.7569	0.7263	0.7211
	F-measure	0.7545	**0.7681**	0.7541	0.7268	0.7359
DS3	Accuracy	0.4123	0.4203	0.4210	**0.4268**	0.4238
	Precision	0.3713	**0.3736**	0.3533	0.3658	0.3527
	Recall	0.3609	0.3551	0.3635	**0.3695**	0.3607
	F-measure	0.3648	0.3677	0.3569	**0.3690**	0.3553
DS4	Accuracy	0.5146	0.5156	0.5235	**0.5252**	0.5281
	Precision	0.5495	0.5372	0.5566	**0.5579**	0.5567
	Recall	**0.5576**	0.5050	0.5185	0.4962	0.5171
	F-measure	0.5384	0.5205	0.5366	**0.5385**	0.5360
DS5	Accuracy	**0.5124**	0.4705	0.4809	0.4743	0.4786
	Precision	0.4131	0.4067	0.4112	**0.4176**	0.4080
	Recall	0.4038	0.3826	0.3933	**0.4036**	0.4023
	F-measure	0.4098	0.3939	0.4013	**0.4156**	0.4042
DS6	Accuracy	**0.4915**	0.4855	0.4575	0.4432	0.4665
	Precision	0.4421	0.4403	0.4313	**0.4434**	0.4338
	Recall	0.4301	0.4444	0.4347	**0.4497**	0.4466
	F-measure	0.4331	0.4456	0.4326	**0.4461**	0.4394
DS7	Accuracy	**0.3676**	0.3535	0.3445	0.3558	0.3579
	Precision	0.3761	0.3721	0.3622	0.3638	**0.3790**
	Recall	**0.3418**	0.3309	0.3186	0.3334	0.3303
	F-measure	0.3508	0.3501	0.3385	0.3476	**0.3527**
DS8	Accuracy	0.7295	0.7354	0.7400	**0.7402**	0.7321
	Precision	0.7754	0.7699	0.7800	**0.7841**	0.7741
	Recall	0.7410	0.7242	0.7399	**0.7450**	**0.**7352
	F-measure	0.7654	0.7605	0.7625	**0.7658**	**0.**7606
DS9	Accuracy	0.3854	0.3877	0.3899	**0.3910**	0.3856
	Precision	0.3814	0.3865	0.3847	**0.3921**	0.3910
	Recall	0.3951	0.3895	0.3995	**0.4010**	0.4009
	F-measure	0.3855	0.3821	0.3895	**0.3988**	0.3951
Summation		5	4	0	25	2

7.4. Tuning parameters of the krill herd algorithm

The experimental design aims to examine the behavior of the proposed KH algorithm with application to text document clustering problem. These experiments assist in the tuning values of all independent parameters of the KHA, and thus the optimal values of the KH parameters can be selected. Table 3 shows a set of experiments were conducted using 20 convergence scenarios, each of which has different system parameter settings. These experiments determine the suitability of the KH algorithm for solving the text document clustering problem. These experiments study four different parameters, namely, KHM size (S), foraging activity (V_f), physical diffusion (D_{max}), and movement induced (N_{max}). Each parameter is studied by keeping the other parameters constant. The maximum number of iterations ($Imax = 1000$) is fixed for all experiments (Forsati et al., 2008; Bharti and Singh, 2016a; Abualigah et al., 2017c). Each convergence scenario on all original nine datasets was experimented and analyzed.

Table 3 shows the four divisions of convergence scenarios. One parameter in each part is manipulated to choose the optimal value among the five values. For example, in scenario (6) to scenario (10), three parameters are fixed (i.e., S, D_{max}, and N_{max}) and one parameter is manipulated (i.e., V_f). The first part examines the KHM size (S) with five different values (i.e., $S = 10, S = 20, S = 30, S = 40$, and $S = 50$), which is the number of KH solutions. The second part examines the foraging activity (V_f) (i.e., $V_f = 0.005, V_f = 0.010, V_f = 0.030, V_f = 0.040$, and $V_f = 0.070$). The third part examines the physical diffusion (D_{max}) (i.e., $D_{max} = 0.008, D_{max} = 0.030, D_{max} = 0.040, D_{max} = 0.080$, and $D_{max} = 0.100$). Finally, the fourth part examines the movement induced (N_{max}) (i.e., $N_{max} = 0.005$, $N_{max} = 0.020$, $N_{max} = 0.030$, $N_{max} = 0.050$, and $N_{max} = 0.100$). All of these parameter values have been determined according to the most related works in the literature of the KH algorithm (Gandomi and Alavi, 2012; Bolaji et al., 2016) and TD clustering (Forsati et al., 2008, 2013; Song et al., 2015). The best parameters values (i.e., $S = 20$, $V_f = 0.030$, $D_{max} = 0.008$ and $N_{max} = 0.100$ as shown in Table 3) are obtained by scenario (20). Which obtained best results 24 out of 32.

This section provides a comprehensive discussion of BKHA behavior when solving TDCP using the 20 different scenarios. The discussion considers the behavior of BKHA based on its four main parameter settings.

Scenarios 1 to 5 is used to determine the optimal value for the KHM size (i.e., $S = 10$, $S = 20$, $S = 30$, $S = 40$, and $S = 50$), which means that the parameters of BKHA are taken as $S = 10$, 20, 30, 40 and 50, $V_f = 0.020$, $D_{max} = 0.002$ and $N_{max} = 0.010$. The second scenario in this part obtained 22 best results out of 32 in terms of the evaluation measures as shown in Table 4. The best value for S is 20, which will be used in the remaining scenarios.

Scenarios 6 to 10 is used to determine the optimal value for the foraging activity V_f (i.e., 0.005, 0.010, 0.030, 0.040, and 0.070). The value for

Table 7
The results of BKHA convergence scenarios (Scenario (16) through Scenario (20)).

Dataset	Measure	Scenario number				
		16	17	18	19	20
DS1	Accuracy	0.4500	0.4560	0.4729	0.4722	**0.5055**
	Precision	0.4342	0.4422	0.4667	0.4613	**0.5084**
	Recall	0.3991	0.4031	0.4371	0.4356	**0.4747**
	F-measure	0.4151	0.4210	0.4502	0.4472	**0.4901**
DS2	Accuracy	0.7507	0.7543	0.7645	0.7372	**0.7798**
	Precision	0.7273	0.7689	0.7747	0.7465	**0.7834**
	Recall	0.7263	0.7508	0.7580	0.7327	**0.7738**
	F-measure	0.7268	0.7596	0.7662	0.7395	**0.7786**
DS3	Accuracy	0.3997	0.4166	0.4204	0.3879	**0.4255**
	Precision	0.3617	0.3528	0.3543	0.3421	**0.3676**
	Recall	0.3587	0.3639	0.3658	0.3264	**0.3586**
	F-measure	0.3583	0.3566	0.3586	0.3321	**0.3623**
DS4	Accuracy	**0.5341**	0.5023	0.5182	0.5081	0.5057
	Precision	0.5557	0.5348	0.5402	0.5431	0.5245
	Recall	0.5189	0.4881	0.5056	0.4971	0.4804
	F-measure	**0.5366**	0.5100	0.5220	0.5187	0.5014
DS5	Accuracy	0.4759	0.4753	0.4803	0.4493	**0.4861**
	Precision	**0.4104**	0.4170	0.3868	0.4083	**0.4281**
	Recall	**0.3849**	0.3948	0.4078	0.3599	**0.4145**
	F-measure	0.3960	0.4018	0.4115	0.3722	**0.4207**
DS6	Accuracy	0.4608	0.4595	0.4538	0.4241	**0.4629**
	Precision	**0.4336**	0.4286	0.4123	0.4025	0.4314
	Recall	0.4493	0.4394	0.4284	0.4083	**0.4523**
	F-measure	0.4355	0.4333	0.4198	0.4047	**0.4461**
DS7	Accuracy	0.3370	0.3452	**0.3669**	0.3546	0.3370
	Precision	0.3481	0.3607	**0.3685**	0.3679	0.3415
	Recall	0.3134	0.3196	0.3402	0.3269	**0.3421**
	F-measure	0.3294	0.3385	**0.3536**	0.3458	0.3459
DS8	Accuracy	0.7321	0.7366	0.7413	0.7401	**0.7423**
	Precision	0.7760	0.7741	0.7812	**0.7851**	0.7764
	Recall	0.7401	0.7332	0.7401	0.7449	**0.7454**
	F-measure	0.7650	0.7616	0.7612	0.7625	**0.7650**
DS9	Accuracy	0.3894	0.3931	0.3920	0.3861	**0.3951**
	Precision	0.3912	0.3948	0.4011	0.3958	**0.4098**
	Recall	0.4055	0.4112	0.4067	0.3996	**0.4147**
	F-measure	0.4001	0.4057	04032	0.3971	**0.4102**
Summation		5	0	3	1	27

KHM is fixed based on the previous scenarios while the physical diffusion D_{max} and the movement induced by other krill individuals N_{max} are fixed based on previous literature. The eighth scenario obtained 15 best results out of 32 based on the evaluation measures as shown in Table 5. The best value for V_f is 0.030, which will be used in the remaining scenarios.

Scenarios 11 to 15 is used to determine the optimal value for the physical diffusion D_{max} (i.e., 0.008, 0.030, 0.040, 0.080, and 0.100). The values for KHM and foraging activity are fixed based on the previous scenarios while the movement induced by other krill individuals N_{max} is fixed based on previous literature. The eleven scenario obtained 15 best results out of 32 based on the evaluation measures as shown in Table 6. The best value for D_{max} is 0.008, which will be used in the remaining scenarios.

Scenarios 16 to 20 is used to determine the optimal value for the movement induced by other krill individuals N_{max} (i.e., 0.005, 0.020, 0.030, 0.050, and 0.100). The values for KHM, foraging activity, and physical diffusion are fixed based on the previous scenarios. The twentieth scenario in this part obtained 24 best results out of 32 based on the evaluation measures as shown in Table 7. The best value for N_{max} is 0.100. Finally, the best parameters values are $S = 20$, $V_f = 0.030$, $D_{max} = 0.008$, and $N_{max} = 0.100$ were obtained by Scenario (20).

7.5. Results and discussion

The experiment was evaluated in terms of algorithm performance and compared with similar algorithms.

Table 8 shows the results on nine text document benchmark datasets, summarized based on the criteria of accuracy, precision, recall, and *F*-measure. The best results for each dataset among the evaluation measures are highlighted in **bold** font.

In this section, the correctness of the document clusters created using the nine clustering algorithms was compared across nine various text datasets. Table 8 shows the first part of the results of the text document clustering algorithms on the basis of the evaluation measures. According to the accuracy measure, the proposed MHKHA showed the highest algorithm performance among all the nine clustering algorithms. This algorithm achieved the best recorded results of nine out of nine datasets. Given this accuracy measure, which is a common evaluation measure employed in the text clustering domain, we evaluated and compared the proposed clustering algorithms with the most successful method in the text clustering domain using the same datasets. The hybrid with the *k*-mean algorithm was more effective than the original KH algorithms and other similar clustering algorithms.

Table 8 also shows the clustering results in terms of the precision and recall measures. Both parameters are based on judgment and measure the number of truly relevant documents. The proposed MHKHA was ranked the highest on the basis of these two measures (i.e., precision and recall measures). The algorithm obtained the best results in nine out of nine datasets in terms of precision measure. The algorithm was followed by HKHA1, HKHA2, and HKHA3. Given these measures, which are important significant evaluation measures in the text clustering domain, the proposed HKHAs, particularly, HKHA1, achieved the most successful performance outcomes among those of all the comparative hybrid algorithms tested.

Lastly, *F*-measure is a standard evaluation measure generally used in text mining, particularly in the text clustering domain, to test clustering accuracy. This measure is determined by calculating the score of the correct results divided by the number of all positive results. The proposed MHKHA was also ranked as the highest-performing clustering algorithm. The algorithm obtained the best results in nine out of nine datasets. The second best was HKHA1. These results are consistent with the precision and recall evaluation measures, further supporting the proposed algorithms, particularly MHKHA. MHKHA is an effective and efficient algorithm for solving text document clustering problems compared with other similar algorithms and the original KH algorithms. Therefore, deriving initial solutions from the *k*-mean using the proposed combination of objective functions enhances the local search ability because the *k*-mean is considered a strong local search algorithm for the clustering technique. The *k*-mean also improves global search ability while powerful solutions begin the KH algorithm.

7.6. Statistical significance analysis

The Friedman test ranking is performed based on the *F*-measure values to evaluate the algorithm ranking as shown in Table 9 (Bharti and Singh, 2016a; Nayak et al., 2017; Abualigah et al., 2017c, b). The proposed MHKHA is ranked the highest over nine methods to improve the text document clustering, followed by HKHA1, HKHA3, HKHA2, HPSO, HHS, HGA, KHA2, HKA1, and *K*-mean++. MHKHA is an effective algorithm to solve the text document clustering problem based on the average ranking analyses because of the initial solutions from the *k*-mean algorithm using the combination of objective functions that enhances the local search ability. This combination caused the *k*-mean clustering algorithm to find the local optimum solution accurately and therefore enhance the performance of the KH algorithm.

The performance of the proposed methods (i.e., basic KH (KHA1), hybrid KH (MKHA1), and combination of objective functions and hybrid KH (MHKHA)) are further evaluated using the *t*-test on the nine datasets

Table 8
Comparison of clustering algorithm performance.

Measure	DS	HHS	HGS	HPSO	*K*-mean++	KHA1	KHA2	HKHA1	HKHA2	HKHA3	MHKHA
Accuracy	DS1	0.5548	0.6112	0.5675	0.5452	0.5794	0.5550	0.6122	0.5642	**0.6038**	0.5585
	DS2	0.7716	0.7260	0.7513	0.7436	0.7617	0.7318	0.8737	0.8354	0.8459	**0.9483**
	DS3	0.4313	0.4397	0.4352	0.3997	0.4181	0.4264	0.4622	0.4441	0.4132	**0.4638**
	DS4	0.5541	0.5519	0.5719	0.5582	0.4980	0.5348	0.5846	0.5682	0.5782	**0.6111**
	DS5	0.5641	0.6200	0.5719	0.5679	0.5699	0.5522	0.5762	0.5682	0.5752	**0.5826**
	DS6	0.5863	0.5889	0.5812	0.5772	0.5941	0.5859	0.6067	0.5834	0.6029	**0.6366**
	DS7	0.5484	0.5261	0.5470	0.5202	0.5543	0.5636	0.5681	0.5414	0.5422	**0.5779**
	DS8	0.7541	0.7540	0.7610	0.7450	0.7423	0.7358	**0.7689**	0.7582	0.7568	0.7688
	DS9	0.4297	0.4315	0.4218	0.4251	0.3951	0.3904	0.4301	0.4215	0.4018	**0.4557**
Precision	DS1	0.5640	0.5608	0.5544	0.4305	0.5049	0.5478	**0.6670**	0.5534	0.5972	0.5966
	DS2	0.7257	0.6461	0.6726	0.7056	0.7031	0.6764	0.8714	0.8237	0.8345	**0.9573**
	DS3	0.3483	0.3669	0.3768	0.3617	0.3641	0.3800	0.3807	0.3786	0.3751	**0.4201**
	DS4	0.5341	0.5178	0.5327	0.5224	0.5584	0.5183	0.5717	0.5687	0.5421	**0.5909**
	DS5	0.4646	0.5029	0.5030	0.4809	0.4757	0.4700	0.4715	0.5097	0.5030	**0.5259**
	DS6	0.4911	0.4896	0.4947	0.4685	0.4985	0.4735	0.5167	0.4815	0.5093	**0.5188**
	DS7	0.5060	0.4814	0.5116	0.4462	0.5080	0.5112	0.5226	0.5109	0.5117	**0.5605**
	DS8	0.7854	0.7863	0.7741	0.7789	0.7764	0.7710	0.7985	0.7985	0.7785	**0.8081**
	DS9	0.4059	0.4154	0.3954	0.4114	0.4098	0.3987	0.4354	0.4278	0.4029	**0.4654**
Recall	DS1	0.5358	0.5367	0.5390	0.5096	0.5091	0.5162	0.5982	0.5549	0.5888	**0.5998**
	DS2	0.7328	0.6756	0.7100	0.7295	0.7212	0.6881	0.8616	0.8223	0.8301	**0.9390**
	DS3	0.3405	0.3616	0.3826	0.3587	0.3682	0.3792	0.3845	0.3754	0.3630	**0.4144**
	DS4	0.5249	0.5308	0.3505	0.5311	0.5480	0.5257	0.5711	0.5876	0.5575	**0.5806**
	DS5	0.4496	0.4845	0.4777	0.4670	0.4495	0.4374	0.4561	0.4827	0.4919	**0.5236**
	DS6	0.5338	0.5082	0.5345	0.5043	0.5316	0.5001	0.5484	0.5166	0.5395	**0.5494**
	DS7	0.5080	0.4927	0.5110	0.4498	0.5159	0.5328	0.5288	0.5060	0.5160	**0.5555**
	DS8	0.7455	0.7548	0.7541	0.7659	0.7454	0.7411	0.7514	0.7511	0.7488	**0.7674**
	DS9	0.3954	0.4157	0.4245	0.4284	0.4147	0.4015	0.4301	0.4106	0.4178	**0.4558**
F-measure	DS1	0.5485	0.5468	0.5452	0.5152	0.5010	0.5301	0.5798	0.5529	0.5700	**0.5994**
	DS2	0.7288	0.6578	0.6898	0.7171	0.7112	0.6817	0.8664	0.8229	0.8318	**0.9481**
	DS3	0.3434	0.3634	0.3779	0.3583	0.3646	0.3779	0.3806	0.3758	0.3679	**0.4245**
	DS4	0.5231	0.4082	0.5491	0.5285	0.4980	0.5524	0.5984	0.5706	0.5775	**0.5996**
	DS5	0.4562	0.5027	0.4895	0.4731	0.4618	0.4525	0.4631	0.4955	0.5069	**0.5236**
	DS6	0.5105	0.4982	0.5129	0.4810	0.5138	0.4858	0.5315	0.4974	0.5227	**0.5331**
	DS7	0.5070	0.4881	0.5111	0.4480	0.5117	0.5218	0.5320	0.5082	0.5137	**0.5578**
	DS8	0.7421	0.7741	0.7789	0.7652	0.7650	0.7622	0.7812	0.7789	0.7801	**0.7868**
	DS9	0.4245	0.4151	0.4214	0.4201	0.4069	0.4009	0.4332	0.4126	0.4075	**0.4526**

Table 9
The average ranking of the proposed clustering algorithms based on the *F*-measure. i.e., lower rank value is the best method.

Method No.	Method name	Datasets									Mean rank	Final ranking
		#1	#2	#3	#4	#5	#6	#7	#8	#9		
1	HHS	5	5	10	8	9	6	8	10	3	7.11	6
2	HGA	6	10	8	10	3	7	9	6	6	7.22	7
3	HPSO	7	8	3	6	5	5	6	5	4	5.44	5
4	*K*-mean++	9	6	9	7	6	10	10	7	5	7.66	10
5	KHA1	10	7	7	9	8	4	5	8	9	7.44	9
6	KHA2	8	9	3	5	10	9	3	9	10	7.33	8
7	HKHA1	2	2	2	2	7	2	2	2	2	2.55	2
8	HKHA2	4	4	5	4	4	8	7	4	7	5.22	4
9	HKHA3	3	3	6	3	2	3	4	3	8	3.88	3
10	MHKHA	1	1	1	1	1	1	1	1	1	1.00	**1**

through twenty runs as shown in Tables 10 and 11. The significance differences in the results are evaluated using the *t*-test with α <0.05.

Table 10 summarizes the *t*-test results between the KHA1 and HKHA1. A significant improvement in seven out of nine datasets (i.e., DS1, DS2, DS3, DS4, DS7, DS8, and DS9) can be observed. In other cases, improvements are observed but is not significant according to the *t*-test results. In general, the experimental results indicate that HKHA1 outperforms KHA1.

Table 11 summarizes the *t*-test results between HKHA1 and MHKHA. A significant improvement can be observed in the results for the most of the datasets (i.e., DS1, DS2, DS3, DS5, DS7, and DS9). In other cases (i.e., DS4, DS6, and DS8), improvement is observed but is not significant according to the *t*-test results. In general, the experimental results indicate that MHKHA outperforms HKHA1. In the next section, MHKHA is compared with the other algorithms published in the literature.

Table 10
Significance tests of the basic KHA and the hybrid KHA using *t*-test with $\alpha < 0.05$. Highlight (bold) denote that result is significantly different.

Methods			
Datasets	KHA1	HKHA1	*p*-value
DS1	0.5010	0.5798	**3.215e−2**
DS2	0.7112	0.8664	**3.4998e−3**
DS3	0.3646	0.3806	**4.541e−2**
DS4	0.4980	0.5984	**2.1499e−3**
DS5	0.4618	0.4631	2.5963e−1
DS6	0.5138	0.5315	9.541e−2
DS7	0.5117	0.5320	**3.9851e−2**
DS8	0.7650	0.7812	**4.7541e−2**
DS9	0.4069	0.4332	**4.555e−2**

Table 11
Significance tests of the hybrid KHA and the combination of objective functions and hybrid KHA using t-test with $\alpha < 0.05$. Highlight (bold) denote that result is significantly different.

Methods			
Datasets	HKHA1	MHKHA	p-value
DS1	0.5798	0.5994	**4.8691e−2**
DS2	0.8664	0.9481	**5.654e−3**
DS3	0.3806	0.4245	**2.586e−2**
DS4	0.5984	0.5996	3.4512e−1
DS5	0.4631	0.5236	**4.2159e−2**
DS6	0.5315	0.5331	5.321e−1
DS7	0.5320	0.5578	**4.745e−2**
DS8	0.7812	0.7868	4.215-1
DS9	0.4332	0.4526	**3.9119e−2**

7.7. Convergence analysis

This section shows the convergence behavior of the clustering algorithms to illustrate the performance of various proposed KH algorithms compared with other comparative clustering algorithms. The other criterion for evaluating the clustering algorithm is their convergence speed to find the optimal solution (accurate clusters) (Zhou et al., 2017).

Fig. 3 explains the convergence behaviors of hybrid GA, HS, PSO, and KH algorithms under the seven text datasets. Twenty runs were carried out for each dataset, and the average value was estimated based on the convergence behavior (speed) of each clustering algorithm. Moreover, the proposed MHKHA avoided trapping and pursued the global optimum perfectly unlike the other hybrid versions of KH algorithm (i.e., HKHA1, HKHA2, and HKHA3) and other comparative algorithms (i.e., HGA, HHS, and HPSO), which converge to the optimal solution accurately.

Furthermore, the convergence behaviors of HKHA2 and HKHA3 were both faster than those of the HKHA1 version and the comparative algorithms (e.g., HGA, HHS, and HPSO). The proposed hybrid algorithms (e.g., HKHA1, HKHA2, and HKHA3) almost obtained the best solution in comparison with other hybrid algorithms. Hence, the proposed hybrid KH versions demonstrated competitive performances in terms of convergence speed and effectiveness. MHKHA is more effective and efficient than other comparative clustering algorithms in terms of computational time to reach the optimal global solution. This algorithm also produced more accurate document clusters than did the all comparative algorithms. Our study revealed that MHKHA exhibited constant progress throughout the execution. Among all the competitive clustering algorithms. The fitness function values of the MHKHA attained a smooth curve from the first values to the final value (optimum solution) with no sharp changes.

Table 12
Key to the comparator methods.

No.	Method	Key	Reference
01	Combination of Objective Functions and Hybrid Krill Herd Algorithm	MHKHA	Our best
02	Chaotic Gradient Artificial Bee Colony	ABC	Bharti and Singh (2016a)
03	K-mean Text Document Clustering	BPSO + K-mean	Bharti and Singh (2016b)
04	K-mean Text Document Clustering	TV-DF + K-mean	Bharti and Singh (2015)
05	Particle Swarm Optimization	KPSO	Karol and Mangat (2013)
06	Firefly Algorithm	GF-CLUST	Mohammed et al. (2016)
07	Firefly Algorithm	WFA2	Mohammed et al. (2015)
08	Cuckoo Search Algorithm	CS	Zaw and Mon (2015)
09	Heuristic K-means	HK-means	Singh et al. (2011)
10	Bisecting K-mean Algorithm	BK-mean	Prakash et al. (2014)
11	Hierarchical Clustering Algorithm	Hier	Rose (2016)
12	Hybrid Harmony Search with K-mean Algorithm	HS + K-mean	Forsati et al. (2013)
13	An Improved Bee Colony Optimization Algorithm	IBCOCLUST	Forsati et al. (2015)

Table 13
Description of text document datasets that used by the comparative methods.

Dataset	Source	# of documents	# of clusters	Reference
C-DS01	Reuters-21,578	1339	08	Bharti and Singh (2016a), Bharti and Singh (2016b) and Bharti and Singh (2015)
C-DS02	Classic4	2000	04	Bharti and Singh (2016a), Bharti and Singh (2016b) and Bharti and Singh (2015)
C-DS03	WebKB	2803	04	Bharti and Singh (2016a), Bharti and Singh (2016b) and Bharti and Singh (2015)
C-DS04	20Newsgroups	2000	11	Karol and Mangat (2013)
C-DS05	Reuters-21587	1000	05	Karol and Mangat (2013)
C-DS06	20Newsgroups	300	03	Mohammed et al. (2016) and Mohammed et al. (2015)
C-DS07	Reuters-21578	300	06	Mohammed et al. (2016)
C-DS08	TREC	414	09	Prakash et al. (2014) and Mohammed et al. (2016)
C-DS09	TREC	313	08	Prakash et al. (2014)
C-DS10	20Newsgroups	300	03	Mohammed et al. (2015)
C-DS11	WebACE	300	03	Zaw and Mon (2015)
C-DS12	Reuters-21587	1049	10	Prakash et al. (2014)
C-DS13	Classic4	500	04	Prakash et al. (2014)
C-DS14	Classic4	1000	04	Prakash et al. (2014)
C-DS15	20Newsgroups	500	20	Prakash et al. (2014)
C-DS16	20Newsgroups	1000	20	Prakash et al. (2014)
C-DS17	TREC	204	06	Singh et al. (2011)
C-DS18	TREC	927	07	Singh et al. (2011)
C-DS19	TREC	690	10	Singh et al. (2011)
C-DS20	Reuters-21587	200	10	Singh et al. (2011)
C-DS21	20Newsgroups	2000	20	Singh et al. (2011)
C-DS22	Reuters-21587	180	06	Rose (2016)
C-DS23	TREC	873	08	Forsati et al. (2013) and Forsati et al. (2015)
C-DS24	DMOZ	697	14	Forsati et al. (2013) and Forsati et al. (2015)
C-DS25	20Newsgroups	9249	10	Forsati et al. (2013) and Forsati et al. (2015)
C-DS26	WebAce	1560	20	Forsati et al. (2013) and Forsati et al. (2015)

Table 14
A comparison of the results obtained by MHKHA and best-published results.

Dataset	Method	Precision	Recall	*F*-measure	Purity	Entropy
C-DS01	1	0.4871	**0.5357**	**0.5100**	–	–
	2	0.5392	0.3224	0.4022	–	–
	3	0.5841	0.3760	0.4550	–	–
	4	**0.6752**	0.3790	0.4855	–	–
C-DS02	1	0.8081	0.7674	0.7868	–	–
	2	**0.8881**	**0.8254**	**0.8080**	–	–
	3	0.6161	–	0.6716	–	–
	4	0.8798	0.8065	0.8416	–	–
C-DS03	1	**0.4875**	**0.5146**	**0.5155**	–	–
	2	0.4562	0.3503	0.3948	–	–
	3	0.3926	0.3668	0.3820	–	–
	4	0.4726	0.3340	0.3914	–	–
C-DS04	1			**0.4982**	–	**0.3255**
	5	–	–	0.4800	–	0.3400
C-DS05	1	–	–	**0.6271**	–	**0.4238**
	5	–	–	0.3200	–	0.4700
	9	–	–	–	–	0.5350
C-DS06	1	–	–	**0.8463**	**0.9319**	**0.2558**
	6	–	–	0.5218	0.5667	1.3172
C-DS07	1	–	–	**0.6861**	**0.7747**	**0.5264**
	6	–	–	0.3699	0.4867	1.6392
C-DS08	1	–	–	**0.5236**	**0.7332**	**0.4044**
	6	–	–	0.3213	0.4710	2.0119
	10	–	–	0.2478	0.4850	1.4102
C-DS09	1	–	–	**0.5996**	**0.8463**	**0.6188**
	6	–	–	0.3851	0.4920	1.1246
	10	–	–	0.1946	0.3514	1.7344
C-DS10	1	–	–	0.5623	0.7490	**0.3676**
	7	–	–	**0.5753**	**0.7655**	0.8118
C-DS11	1	–	–	**0.7821**	**0.7692**	0.7566
	8	–	–	0.7110	0.6910	**0.6730**
C-DS12	1	–	–	–	**0.7999**	–
	9	–	–	–	0.4000	–
C-DS13	1	–	–	–	**0.7429**	–
	9	–	–	–	0.4760	–
C-DS14	1	–	–	–	**0.9915**	–
	9	–	–	–	0.5130	–
C-DS15	1	–	–	–	0.3871	–
	9	–	–	–	**0.3970**	–
C-DS16	1	–	–	–	**0.5388**	–
	9	–	–	–	0.4000	–
C-DS17	1	–	–	**0.4045**	**0.6552**	**0.4170**
	10	–	–	0.1719	0.4853	1.3351
C-DS18	1	–	–	**0.4352**	–	**0.2944**
	10	–	–	0.1407	–	0.4344
C-DS19	1	–	–	**0.3645**	**0.4404**	**0.5208**
	10	–	–	0.2627	0.4210	1.5922
C-DS20	1	–	–	**0.3515**	0.2471	**0.5487**
	10	–	–	0.2444	**0.2518**	1.9981
C-DS21	1	–	–	**0.3892**	**0.5511**	**0.8709**
	10	–	–	0.1894	0.2141	2.2575
C-DS22	1	–	–	**0.6326**	–	–
	11	–	–	0.2800	–	–
C-DS23	1	–	–	0.8641	**0.8141**	**0.4120**
	12	–	–	**0.8968**	0.7611	0.4401
	13	–	–	0.8826	–	–
C-DS24	1	–	–	**0.8851**	**0.8134**	**0.3199**
	12	–	–	0.8397	0.7328	0.3312
	13	–	–	0.8574	–	–
C-DS25	1	–	–	**0.8089**	**0.7113**	**0.4748**
	12	–	–	0.7805	0.6712	0.5083
	13	–	–	0.7902	–	–
C-DS26	1	–	–	**0.7851**	**0.6248**	**0.4256**
	12	–	–	0.7662	0.5647	0.4701
	13	–	–	0.7342	–	–
Best results		**1/3**	**2/3**	**18/20**	**16/19**	**16/17**

7.8. Comparison with previous methods

The results of the proposed MHKHA were compared with most of the related works using their datasets. Table 12 includes 13 clustering methods using 26 datasets. Table 13 shows the text datasets utilized in this comparison to compare with the other previous methods. This table provides information on each dataset based on the number of documents, number of clusters, and research authors.

Table 14 shows the results of the MHKHA compared with other clustering methods based on several evaluation measures used in the domain of TC, including precision, recall, *F*-measure, entropy, purity, and accuracy. This table is categorized into 26 comparisons based on their benchmark text datasets. Each row includes the research result along with the values of the evaluation measures. The indicator "-" represents where the method did not obtain a result according to the current evaluation measure. The best results based on the overall evaluation measures are the highest except for the entropy measure where the best result is the lowest.

MHKHA achieved the best results in one out of three cases according to the precision measure, in two out of three cases according to the recall measure, in 18 out of 21 cases according to the *F*-measure measure, in 16 out of 19 cases according to the purity measure, and in 16 out of 17 cases according to the entropy measure. The results obtained by the MHKHA overcome the comparative methods conducted by many well-known researchers who have made a significant effort to obtain the best clustering results.

8. Conclusion and future works

Meta-heuristic algorithms have been successfully used to solve many complex optimization problems. The KH algorithm is a recent meta-heuristic optimization algorithm has been proposed for solving several complex global optimization problems. In this paper, a combination of objective functions with the hybrid KH algorithm was proposed to solve text document clustering problem. The original KH algorithm was quickly saturated and subsequently trapped in the local search. However, an enhanced KH algorithm was obtained by introducing the local exploitation of the *k*-mean to avoid trapping in the local optimum and premature convergence. Under this hybridization strategy, HKHAs quickly converged to optimal solutions under KHM initialization by the *k*-mean results. Hybrid versions of the KH algorithm aimed to solve local search problems. As well, a combination of objective functions was introduced into the best hybrid KH version (KH1) to improve the local search ability by obtaining an accurate clustering decisions.

To evaluate the performance of the proposed clustering algorithms, five measures, namely, accuracy, precision, recall, *F*-measure, and fitness function, were used. These assessment measures are the latest evaluation criteria employed in the text clustering domain to evaluate proposed clustering methods. The proposed HK algorithms obtained the best recorded results for all the standard datasets used among all versions of the KH algorithm tested and several successful text document clustering methods from literature. Thus, MHKHA is efficient and effective methods for solving the text document clustering problem. The comparison with other published clustering algorithms in the literature for solving text document clustering problem revealed that the proposed MHKHA is fast and efficient in solving text document clustering problem.

Overall, MHKHA is a suitable algorithmic version for the text document clustering domain. In the future, the algorithm can be applied to other clustering problems to ensure its capability in this domain. Furthermore, the number of clusters can be study to set dynamically in the next study.

References

Abualigah, L.M.Q., Hanandeh, E.S., 2015. Applying genetic algorithms to information retrieval using vector space model. Int. J. Comput. Sci. Eng. Appl. 5 (1), 19.

Abualigah, L.M., Khader, A.T., 2017. Unsupervised text feature selection technique based on hybrid particle swarm optimization algorithm with genetic operators for the text clustering. J. Supercomput. 1–23.

Abualigah, L.M., Khader, A.T., Al-Betar, M.A., 2016a. Multi-objectives-based text clustering technique using *k*-mean algorithm. In: Computer Science and Information Technology, CSIT, 2016 7th International Conference on. IEEE, pp. 1–6.

Abualigah, L.M., Khader, A.T., Al-Betar, M.A., Alomari, O.A., 2017a. Text feature selection with a robust weight scheme and dynamic dimension reduction to text document clustering. Expert Syst. Appl. 84, 24–36.

Abualigah, L.M., Khader, A.T., Al-Betar, M.A., Awadallah, M.A., 2016b. A krill herd algorithm for efficient text documents clustering. In: Computer Applications & Industrial Electronics, ISCAIE, 2016 IEEE Symposium on. IEEE, pp. 67–72.

Abualigah, L.M., Khader, A.T., Al-Betar, M.A., Hanandeh, E.S., 2017. A new hybridization strategy for krill herd algorithm and harmony search algorithm applied to improve the data clustering. Management 9, 11.

Abualigah, L.M., Khader, A.T., Hanandeh, E.S., 2018. A hybrid strategy for krill herd algorithm with harmony search algorithm to improve the data clustering **1**. In: Intelligent Decision Technologies, Preprint, pp. 1–12.

Abualigah, L.M., Khader, A.T., Hanandeh, E.S., 2017b. A new feature selection method to improve the document clustering using particle swarm optimization algorithm. J. Comput. Sci..

Abualigah, L.M., Khader, A.T., Hanandeh, E.S., 2018b. A novel weighting scheme applied to improve the text document clustering techniques. In: Innovative Computing, Optimization and Its Applications. Springer, Cham, pp. 305–320.

Abualigah, L.M., Khader, A.T., Hanandeh, E.S., Gandomi, A.H., 2017c. A novel hybridization strategy for krill herd algorithm applied to clustering techniques. Appl. Soft Comput. 60, 423–435.

Al-Sai, Z.A., Abualigah, L.M., 2017, May. Big data and E-government: A review. In: Information Technology, ICIT, 8th International Conference on. IEEE, pp. 580–587.

Alomari, O.A., Khader, A.T., Al-Betar, M.A., Abualigah, L.M., 2017a. Gene selection for cancer classification by combining minimum redundancy maximum relevancy and bat-inspired algorithm. Int. J. Data Min. Bioinform. 19 (1), 32–51.

Alomari, O.A., Khader, A.T., Mohammed, A.A.B., Abualigah, L.M., Nugroho, H., Chandra, G.R., et al., 2017b. MRMR BA: A hybrid gene selection algorithm for cancer classification. J. Theoret. Appl. Inf. Technol. 95 (12).

Alyasseri, Z.A.A., Khader, A.T., Al-Betar, M.A., Abualigah, L.M., 2017, May. ECG signal denoising using β-hill climbing algorithm and wavelet transform. In: Information Technology (ICIT), 2017 8th International Conference on. IEEE, pp. 96–101.

Balabantaray, R.C., Sarma, C., Jha, M., 2015. Document clustering using K-means and K-medoids. ArXiv preprint arXiv:1502.07938.

Bharti, K.K., Singh, P.K., 2015. Hybrid dimension reduction by integrating feature selection with feature extraction method for text clustering. Expert Syst. Appl. 42 (6), 3105–3114.

Bharti, K.K., Singh, P.K., 2016a. Chaotic gradient artificial bee colony for text clustering. Soft Comput. 20 (3), 1113–1126.

Bharti, K.K., Singh, P.K., 2016b. Opposition chaotic fitness mutation based adaptive inertia weight BPSO for feature selection in text clustering. Appl. Soft Comput. 43, 20–34.

Bolaji, A.L.A., Al-Betar, M.A., Awadallah, M.A., Khader, A.T., Abualigah, L.M., 2016. A comprehensive review: Krill Herd algorithm (KH) and its applications. Appl. Soft Comput. 49, 437–446.

Forsati, R., Keikha, A., Shamsfard, M., 2015. An improved bee colony optimization algorithm with an application to document clustering. Neurocomputing 159, 9–26.

Forsati, R., Mahdavi, M., Kangavari, M., Safarkhani, B., 2008, May. Web page clustering using harmony search optimization. In: Electrical and Computer Engineering, 2008 CCECE 2008 Canadian Conference on. IEEE, pp. 001601–001604.

Forsati, R., Mahdavi, M., Shamsfard, M., Meybodi, M.R., 2013. Efficient stochastic algorithms for document clustering. Inform. Sci. 220, 269–291.

Fu, T.C., 2011. A review on time series data mining. Eng. Appl. Artif. Intell. 24 (1), 164–181.

Gandomi, A.H., Alavi, A.H., 2012. Krill herd: a new bio-inspired optimization algorithm. Commun. Nonlinear Sci. Numer. Simul. 17 (12), 4831–4845.

Jaganathan, P., Jaiganesh, S., 2013, December. An improved *k*-means algorithm combined with particle swarm optimization approach for efficient web document clustering. In: Green Computing, Communication and Conservation of Energy, CGCE, 2013 International Conference on. IEEE, pp. 772–776.

Kanimozhi, K.V., Venkatesan, M., 2015. Survey on text clustering techniques. Adv. Res. Electr. Electron. Eng. 2 (12), 55–58.

Karol, S., Mangat, V., 2013. Evaluation of text document clustering approach based on particle swarm optimization. Open Comput. Sci. 3 (2), 69–90.

Moayedikia, A., Jensen, R., Wiil, U.K., Forsati, R., 2015. Weighted bee colony algorithm for discrete optimization problems with application to feature selection. Eng. Appl. Artif. Intell. 44, 153–167.

Mohammed, A.J., Yusof, Y., Husni, H., 2015. Document clustering based on firefly algorithm. J. Comput. Sci. 11 (3), 453.

Mohammed, A.J., Yusof, Y., Husni, H., 2016. GF-CLUST: A nature-inspired algorithm for automatic text clustering. J. Inf. Commun. Technol. 15 (1).

Nanda, S.J., Panda, G., 2014. A survey on nature inspired metaheuristic algorithms for partitional clustering. Swarm and Evolut. Comput. 16, 1–18.

Nayak, J., Naik, B., Behera, H.S., Abraham, A., 2017. Hybrid chemical reaction based metaheuristic with fuzzy c-means algorithm for optimal cluster analysis. Expert Syst. Appl. 79, 282–295.

Prakash, B.R., Hanumanthappa, M., Mamatha, M., 2014. Cluster based term weighting model for web document clustering. In: Proceedings of the Third International Conference on Soft Computing for Problem Solving. Springer, India, pp. 815–822.

Premalatha, K., Natarajan, A.M., 2010a. Hybrid PSO and GA models for document clustering. Int. J. Adv. Soft Comput. Appl. 2 (3), 302–320.

Premalatha, K., Natarajan, A.M., 2010b. Hybrid PSO and GA models for document clustering. Int. J. Adv. Soft Comput. Appl. 2 (3), 302–320.

Rao, R.V., Rai, D.P., Balic, J., 2017. A combination of objective functions algorithm for optimization of modern machining processes. Eng. Appl. Artif. Intell. 61, 103–125.

Rose, J.D., 2016. An efficient association rule based hierarchical algorithm for text clustering. Int. J. Adv. Eng. Tech. 751, 753.

Shehab, M., Khader, A.T., Al-Betar, M.A., Abualigah, L.M., 2017, May. ybridizing cuckoo search algorithm with hill climbing for numerical optimization problems. In: Information Technology, ICIT, 2017 8th International Conference on. IEEE, pp. 36–43.

Singh, V.K., Tiwari, N., Garg, S., 2011, October. Document clustering using *k*-means, heuristic *k*-means and fuzzy c-means. In: Computational Intelligence and Communication Networks, CICN, 2011 International Conference on. IEEE, pp. 297–301.

Song, W., Qiao, Y., Park, S.C., Qian, X., 2015. A hybrid evolutionary computation approach with its application for optimizing text document clustering. Expert Syst. Appl. 42 (5), 2517–2524.

Tunali, V., Bilgin, T., Camurcu, A., 2016. An improved clustering algorithm for text mining: multi-cluster spherical *k*-means. Int. Arab J. Inf. Technol. 13 (1).

Wang, G.G., Gandomi, A.H., Alavi, A.H., Deb, S., 2016. A hybrid method based on krill herd and quantum-behaved particle swarm optimization. Neural Comput. Appl. 27 (4), 989–1006.

Wang, G.G., Gandomi, A.H., Alavi, A.H., Hao, G.S., 2014a. Hybrid krill herd algorithm with differential evolution for global numerical optimization. Neural Comput. Appl. 25 (2), 297–308.

Wang, G., Guo, L., Wang, H., Duan, H., Liu, L., Li, J., 2014b. Incorporating mutation scheme into krill herd algorithm for global numerical optimization. Neural Comput. Appl. 24 (3–4), 853–871.

Wang, J., Yuan, W., Cheng, D., 2015. Hybrid genetic–particle swarm algorithm: An efficient method for fast optimization of atomic clusters. Comput. Theor. Chem. 1059, 12–17.

Zaw, M.M., Mon, E.E., 2015. Web document clustering by using pso-based cuckoo search clustering algorithm. In: Recent Advances in Swarm Intelligence and Evolutionary Computation. Springer International Publishing, pp. 263–281.

Zhou, Y., Zhou, Y., Luo, Q., Abdel-Basset, M., 2017. A simplex method-based social spider optimization algorithm for clustering analysis. Eng. Appl. Artif. Intell. 64, 67–82.