Hasan Rashaideh*, Ahmad Sawaie, Mohammed Azmi Al-Betar, Laith Mohammad Abualigah, Mohammed M. Al-laham, Ra'ed M. Al-Khatib and Malik Braik

A Grey Wolf Optimizer for Text Document Clustering

https://doi.org/10.1515/jisys-2018-0194
Received April 24, 2018.

Abstract: Text clustering problem (TCP) is a leading process in many key areas such as information retrieval, text mining, and natural language processing. This presents the need for a potent document clustering algorithm that can be used effectively to navigate, summarize, and arrange information to congregate large data sets. This paper encompasses an adaptation of the grey wolf optimizer (GWO) for TCP, referred to as TCP-GWO. The TCP demands a degree of accuracy beyond that which is possible with metaheuristic swarm-based algorithms. The main issue to be addressed is how to split text documents on the basis of GWO into homogeneous clusters that are sufficiently precise and functional. Specifically, TCP-GWO, or referred to as the document clustering algorithm, used the average distance of documents to the cluster centroid (ADDC) as an objective function to repeatedly optimize the distance between the clusters of the documents. The accuracy and efficiency of the proposed TCP-GWO was demonstrated on a sufficiently large number of documents of variable sizes, documents that were randomly selected from a set of six publicly available data sets. Documents of high complexity were also included in the evaluation process to assess the recall detection rate of the document clustering algorithm. The experimental results for a test set of over a part of 1300 documents showed that failure to correctly cluster a document occurred in less than 20% of cases with a recall rate of more than 65% for a highly complex data set. The high F-measure rate and ability to cluster documents in an effective manner are important advances resulting from this research. The proposed TCP-GWO method was compared to the other well-established text clustering methods using randomly selected data sets. Interestingly, TCP-GWO outperforms the comparative methods in terms of precision, recall, and F-measure rates. In a nutshell, the results illustrate that the proposed TCP-GWO is able to excel compared to the other comparative clustering methods in terms of measurement criteria, whereby more than 55% of the documents were correctly clustered with a high level of accuracy.

Keywords: Text document clustering, optimization, metaheuristic, grey wolf optimizer, swarm intelligence.

1 Introduction

Document clustering is a fundamental per-cursor process to text clustering used to organize unsupervised documents, text mining, automatic topic extraction, and information retrieval [5]. Data clustering, detection and disease clustering, open source clustering software, clustering the search engine results, time series clustering, and wireless sensor clustering are but a few typical uses of text clustering [31]. Clustering involves

***Corresponding author: Hasan Rashaideh,** Computer Science Department, P.A.B.G Faculty of Information Technology, Al-Balqa Applied University (BAU), P.O. Box 19117, Salt, Jordan, e-mail: rashaideh@bau.edu.jo
Ahmad Sawaie and Malik Braik: Computer Science Department, P.A.B.G. Faculty of Information Technology, Al-Balqa Applied University (BAU), P.O. Box 19117, Salt, Jordan
Mohammed Azmi Al-Betar: Department of Information Technology, Al-Huson University College, Al-Balqa Applied University (BAU), P.O. Box 50, Al-Huson, Irbid, Jordan
Laith Mohammad Abualigah: Faculty of Computer Sciences and Informatics, Amman Arab University, Amman 11953, Jordan
Mohammed M. Al-laham: Department of Management Information Systems, Amman University College, Al-Balqa Applied University (BAU), Amman, Jordan
Ra'ed M. Al-Khatib: Department of Computer Sciences, Faculty of Information Technology and Computer Sciences, Yarmouk University (YU), Irbid, Jordan

splitting a set of homogenous objects into a particular number of clusters. The key idea beyond clustering a set of data is to achieve an inherent structure in the data and to show up this data structure as a set of groups. The data objects within each group must exhibit a high degree of similarity, while the similarity issue should be satisfied among different clusters [27].

Interestingly, text document clustering, as a way of splitting documents into their parent groups, can handle documents with unlabeled or unauthorized clusters. The text clustering process is affected by the calculation of the similarity measure between the documents in each cluster. In general, text clustering is used to improve the reliability of text document organization and provide the user with an understandable view of documents based on an evaluation criterion that can be represented as an objective function. Specifically, text clustering-based schemes have helped users to simultaneously deal with a large number of documents and their organization with greater precision and simplicity than without a text clustering-based model. Text clustering has valuable applications in many domains such as ontology-based text clustering, automatic clustering of newspapers, text categorization, search engine, and clinical work [14, 15, 46].

Most document clustering algorithms can be categorized into two main groups: the partitioning and the hierarchical clustering groups [27]. The hierarchical techniques produce an entangled sequence of division, with single, all-inclusive clusters at the topmost and individual clusters of individual points at the lowermost. The partitioning clustering method aims to partition an aggregation of documents into a set of non-interfering groups, so as to boost the assessment value of clustering. The partitioning clustering method seeks to partition a collection of documents into a set of non-overlapping groups. This is to maximize the evaluation value of clustering and to achieve a high degree of similarity among the clusters. Despite the fact that hierarchical clustering approach is predominantly characterized as a preferable quality clustering approach, this approach does not contain any provision for the reallocation of entities, which may have been incompetently classified in the early stages of text analysis.

More recently, the partitional clustering approach is well-suited for clustering a large collection of document data sets because of its relatively shallow computational effort [50]. Further, the time complexity of most partitional clustering approaches is nearly linear, making them widely practiced. The vector space model (VSM), as a structured model, is used in text clustering to assort the documents. VSM facilitates analysis and data representation. This model is used to represent each document as a vector of frequency term. While each frequency term is represented as a single position, this model assists to identify the distances between the documents and their cluster centroid [13].

For instance, if a document accommodates clusters with settled clusters, it can be categorized as a hierarchical document organization. On the other hand, if the document is empty of any settled clusters, then, it can be rated as a flat document organization. Thus, a reliable and efficient method is needed to establish document organization, either manually or automatically, that can easily be used to organize and manage the vast number of documents in an effective way [9, 13]. Also, a method that is largely invariant to complexity and size of documents within data sets is needed. The goal is to conduct a comprehensive analysis of all information, preferably to cluster a substantial number of text documents in a low computational burden.

Quite recently, text document clustering was formulated in an optimization context aimed at maximizing or minimizing an objective criterion that must be defined for the clustering-based optimization algorithm [4]. To achieve the predefined objective criterion (either minimization or maximization), it is necessary to reach the minimum value of the distance between each document with its cluster centroid. Equivalently, it is also necessary to maximize the objective criterion, and it is essential to decree the largest similarity value between each document and its cluster centroid [29].

The most conventional partitioning clustering algorithm is the K-means algorithm and its alternatives [45]. The benefits of the K-means clustering algorithm are that it is fast, simple, computationally efficient, requires a modest memory, unpretentious, and based on a solid basis for analyzing the variances between clusters. Also, the K-means clustering algorithm is utilized in several meta-heuristic optimization algorithms such as the self-organizing maps (SOM) by Merkl [35] and genetic algorithm (GA) by Raghavan and Birchard [42] were presented in literature to solve the text clustering problem (TCP). The particle swarm optimization (PSO) by Kennedy and Shi [30], as an efficient computational intelligence method, has also been applied to

cluster text documents [19]. Each optimization algorithm, at each iteration, looks for an optimal solution as evaluated by a predetermined fitness function [19].

Recently, several swarm-based algorithms are proposed such as the bat-inspired algorithm [7, 8], grey wolf optimizer (GWO) [37], artificial bee colony [11], flower pollination algorithm [10], etc. A successful and effective swarm-based algorithm that recently proposed to imitate the hunting behavior of the grey wolves is the so-called GWO. The GWO is a simple, population-based, flexible, and derivative-free meta-heuristic optimization method that intelligently avoids stagnation in local optima spots of the search space. It simulates the social behaviors of grey wolves in the aspects of their hierarchical leadership and hunting maneuver. Mirjalili et al. [37] compared the GWO to the state-of-the-art optimization algorithms and showed that the GWO was able to achieve competitive results in terms of exploitation (through the use of unimodal functions), exploration (by means of multimodal functions), local minima avoidance (benchmarked through composite functions), and convergence. The GWO also showed good performance when used in real problems (both constrained and unconstrained) with unknown search domains.

Further, it has many features such as simplicity, flexibility, adaptability, scalability, usability, and stability. These key features have presented the GWO as a very promising approach to being effective for a broad number of real optimization problems as surveyed in Ref. [22]. The GWO is adapted for several optimization applications such as the feature selection [21, 34], Training neural networks [36, 38], clustering applications [16, 47], engineering applications [32, 49], scheduling [33], wireless sensor network [20], environmental modeling applications [41, 44], medical and bioinformatics application [28, 39], image processing [48], and geophysical applications [6]. However, the GWO has not been yet investigated for text document clustering.

Procedurally, this optimization algorithm models the leadership hierarchy of grey wolves, or swarms, by classifying them into four hierarchy social groups according to dominance from top to bottom as α, β, δ, and ω. To elaborate, the α members represent the group of the best solutions in the swarm found for hunting, the β members represent the group of the second best solutions in the swarm, the δ members represent the group of third-best solutions in the swarm, while the ω members represent the rest of the solutions. Using the encircling and attaching pray process, the best three solutions in the α, β, and δ groups will be repeatedly utilized to attract the ω members to better positions in the search space and, thus, drive the search for optimality.

In this work, the remarkable role played by TCP in many areas and the fruitful achievements of the GWO in a wide range of applications have encouraged us to adapt the GWO to address the TCP in an efficient mode, a method referred to as the TCP-GWO. The adaptation involves utilizing and mapping the text clustering concepts into the framework of the GWO. The approach proposed is to randomly select a number of different document vectors from the document collection as the initial cluster centroid vectors. Then, for each grey wolf, each document vector in the document set was assigned to the closest centroid vector based on the average distance of documents to the cluster centroid (ADDC) fitness criterion. The TCP-GWO uses the ADDC as an objective function to repeatedly optimize the distance between the documents and their cluster centroid. Finally, the search for convergence is stopped when the maximum number of iterations is reached or the newly computed average change in centroid vectors is converged, as determined by a predefined threshold value.

The performance of the proposed TCP-GWO method was demonstrated on six text publicly available data sets that were randomly selected with varying sizes and complexity, referred to in the scope of this paper as D1, D2, D3, D4, D5, and D6. The proposed TCP-GWO was devised with extensive experiments and statistical analysis to illustrate the usability and appropriateness of the entire procedures. The results were explored with a discussion using accuracy, precision, recall, and F-measure criteria to assess the degree of performance of the TCP-GWO. The reliability of the presented method was compared with other competitive methods using the same data sets. The comparative results show that the proposed TCP-GWO reveals superior clustering performance and outperforms its competitors.

The remaining parts of this paper are organized as follows: Section 2 introduces a definition and mathematical formulation of the text clustering problem. The grey wolf optimizer for text clustering problem is presented in detail in Section 3. Also, Section 3 provides a general overview of the GWO. The evaluation criteria are presented in Section 4. Section 5 provides the detailed experimental setup and results for comparing

the performance of the GWO-based text clustering method algorithm with the other comparative approaches. The discussion of the experiment's results is also presented in Section 5 with concluding comments and outlooks for further work in Section 6.

2 Preludes for Document Clustering Problem

This section describes how TCP can be formulated in optimization context. This section also provides the solution representation of the TCP encoding modification. The term weighting refers to the term frequency-inverse document frequency (TF-IDF) that was used satisfactorily as an objective function to assess each position. As focusing, the primary motivation of this work is to get an optimal subset of clusters.

2.1 Text Document Formulation and Modeling

The development of text clustering requires some necessary preliminary stages, which should be completed as a pre-step to simplify the text clustering process. This involves data tokenization, removal stop word, and stemming. The data preprocessing methods are required to extract valid text document representation from the available empirical text document data sets.

1. Tokenization is a procedure to break a sequence letter of the document into tokens. A token is any characters compressed between two spaces, such as words, keywords, phrases, symbols, and other elements [43].
2. Stops words would be ostracized. Examples of stop words cover pronouns, punctuation marks, conjunctions, contra-conjunctions, and many more, for example, "a", "against", "about", "am", "all", "above", "after", "and", "again", "any", and "an". A list of stop words containing 571 stop words is publicly available at http://www.unine.ch/Info/clef/.
3. Stemming is the process of removing the prefixes and suffixes from the original words to obtain the basic form of the words. Prefixes are letters inserted at the beginning of the words, while suffixes are letters added at the end of the words. In English specifically, prefixes and suffixes are defined well and can be correctly identified. For example, "ed" is used to refer to the past tense of a verb; also "ing" and etc. are removed. Typically, this process is performed using the Porter Stemmer, which gets rid of some of the extremities such as eliminating prefixes and suffixes from each term such as "ed", "ly", and "ing". For example, the words or terms "connection", "connective", "connected", "connections", "connecting", and "connected" have the mutual root "connect". This root after some of the preprocessing processes will be called term [24].

It is worth mentioning that these preprocessing methods are already utilized in the data set used; thus, the manipulated data sets are in the form of postprocessing.

- The "term weighting", or the TF-IDF as customarily referred to in text clustering, is predominantly used as a standard scheme to allocate a weighting score to each document term using equation 3 [12]. This scheme relies on the TF and IDF to represent each document component [17]. In this search, the TF-IDF mechanism was used to discriminate between the document terms that are applied as an *objective function*. The following assumptions are necessitated to standardize the terms:
- The document set is specified by D as shown below:

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

where n is the number of documents in the documents group D, and d_i is the i^th document, represented mathematically as:

$$d_i = \left[w_{i,1}, w_{i,2}, \ldots, w_{i,t}\right] \tag{2}$$

where i represents the order of the i^th document, d_i represents the document vector of the i^th document, $w_{i,1}, w_{i,2}, \ldots, w_{i,j}, \ldots, w_{i,t}$ is a vector of term weighting, t is the number of distinct terms (vector length), and $w_{i,j}$ represents the weight score of the j^th term for the i^th document, defined in equation (3):

$$\begin{aligned} w_{i,j} &= tf_{i,j} \times idf_{i,j} \\ &= tf_{i,j} \times \log_2 \left(\frac{n}{df_{i,j}} \right) \end{aligned} \tag{3}$$

where $tf_{i,j}$ denotes the number of occurrences of the j^th term in the i^th document, $idf_{i,j}$ denotes a parameter utilized to enhance the terms based on the number of occurrences in the i^th document, n identifies the total number of documents in the collection, and $df_{i,j}$ specifies the term frequency in the collections of documents.

This weighting strategy dilutes the repetitive words with little discernible power.

– The model (matrix) represented in equation (4) describes the documents using the VSM format [3].

$$VSM = \begin{bmatrix} w_{(1,1)} & w_{(1,2)} & \cdots & w_{(1,n)} \\ w_{(2,1)} & w_{(2,2)} & \cdots & w_{(2,n)} \\ \vdots & \vdots & \cdots & \vdots \\ w_{(m,1)} & w_{(m,2)} & \cdots & w_{(m,n)} \end{bmatrix} \tag{4}$$

The ultimate goal to tackle the TCP is to find the optimal subset of clusters, which is defined as follows.

Initially, equation (1) specifies the representation of the document set in the data set. The document group D is divided into K clusters. The centroid must be identified for each cluster as given in equation (5).

$$C = [c_1, c_2, \ldots, c_i, \ldots, c_K] \tag{5}$$

where C is the set of all cluster centroids in the document set, c_i represents the centroid of cluster i, and K is the number of clusters in the document set.

2.2 Objective Function: Similarity Criterion

The similarity between two documents needs to be measured in a clustering analysis. The similarity metric measure computed between documents d_r and d_k is based on the Minkowski distances [18], given by equation (6):

$$D_n(d_r, d_k) = \left[\sum_{i=1}^{d_d} |d_{i,r} - d_{i,k}|^n \right]^{\frac{1}{n}} \tag{6}$$

The Euclidean distance measure can be obtained when $n = 2$. This Euclidean distance is broadly used in text document clustering. In order to manipulate equivalent threshold distances, considering that the distance ranges will vary according to the dimension number, most algorithms use the normalized Euclidean distance as the similarity metric of two documents, d_r and d_k, in the vector space. Equation (7) represents the distance measurement formula based on the normalized Euclidean distance criterion:

$$d(d_r, d_k) = \left[\sum_{s=1}^{d_d} (d_{rs} - d_{ks})^2 / d_d \right]^{1/2} \tag{7}$$

where d_r and d_k are two document vectors; d_d denotes the dimension number of the vector space; d_{rs} and d_{ks} stand for the weight values for the documents d_r and d_k in dimension s.

Table 1: An Example of a Solution Representation of the TCP-GWO.

Document (*d*)	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Cluster(*C*)	C_1	C_5	C_5	C_1	C_2	C_3	C_5	C_8	C_3	C_2
X	1	5	5	1	2	3	5	4	3	2

The average distance of documents to the cluster centroid (ADDC) is used in this paper as an objective function to compute the fitness value (*fit*) and evaluate the solution represented by each wolf.

The fitness values, representing the average distance values from the documents to the cluster centroids, were recalculated inside each iteration loop until convergence. The search for convergence is stopped when the maximum number of iterations was reached or the newly computed fitness value has converged, as defined by the greatest fitness value (the optimal solution).

The fitness value, *fit*, was measured using equation (8) [19]:

$$fit = \left\lfloor \sum_{j=1}^{K} \frac{\left\lceil \frac{\sum_{i=1}^{n} d(c_j, d_{ji})}{r_j} \right\rceil}{K} \right\rfloor \tag{8}$$

where K denotes the number of clusters, d_{ji} denotes the i^th document vector, which belongs to cluster j, c_j is the centroid vector of the j^th cluster; $d(c_j, d_{ji})$ represents the distance between cluster centroid c_j, and the document d_{ji}, r_j refers to the number of documents, which belongs to cluster C_j.

The cluster centroid vector c_j was calculated using equation (9):

$$c_j = \frac{1}{n_j} \sum_{\forall d_j \in c_j} d_j, \tag{9}$$

where c_j represents the centroid vector, d_j represents the document vectors that belong to the j_{th} cluster, and n_j is the number of document vectors that belong to cluster j_{th}.

2.3 Solution Representation

Each solution is represented as a subset of clusters in the range from *1* to *K*. The dataset clustered for the proposed TCP-GWO (to be discussed in the section) was represented as a set of vectors $X = (x_1, x_2, \ldots, x_{i-1}, x_i, x_{i+1}, \ldots, x_{n-1}, x_n)$, where x_i corresponds to a single feature vector, which properly represents an object. Each position denotes one document to which cluster belongs, where the i_{th} position in the solution denotes the situation of the i_{th} document [2]. As aforementioned, the text document solutions were represented using the VSM.

Table 1 shows an illustrative example of a TCP solution representation.

Table 1 exhibits 10 documents and five clusters, where the *X* vector represents a solution, and each value of *X* represents a cluster number. Each document was doled out to a cluster at the end of the text clustering procedure. For example, documents 1 and 4 were grouped into cluster 1, documents 5 and 10 in cluster 2, documents 6 and 9 in cluster 3, document 8 in cluster 4, and finally documents 2, 3, and 7 in cluster 5.

3 Text Clustering Problem-Based Grey Wolf Optimizer

In this section, the TCP-GWO and the theoretical concepts and basic background of GWO are described in detail. Recently resigned, one of most successful swarm-based algorithm is the GWO, which imitated the driving force and hunting performance of the grey wolf packs. Swarm intelligence is capable of keeping information about the search space at each iteration [22, 37].

In the TCP-GWO, the most powerful wolf is called α (i.e. the best text document solution), which redirects the whole pack to hunt, migrate, and feed each other. When the α wolf is not found because of illness, dead,

or for other reasons, another fittest wolf from the β wolves (i.e. the second-best text document solution) will be the leader of the pack. It should be noted that the δ (i.e. the third-best text document solution) and ω members have less power than the α and β as can be seen in Figure 1. This hierarchal social behavior is the strongest point of the GWO algorithm [25, 36].

Another inspiration is the hunting process of grey wolves. When the prey is attacked, a set of efficient steps is followed by grey wolves: attacking, encircling, chasing, and harassing. This enables them to attack strong preys. The following steps in the TCP-GWO are discussed in the subsections below.

3.1 Initialize GWO Parameters

In this step, the parameters of the GWO and the TCP are initialized. The TCP parameters are extracted from the data sets, which are defined in Section 2. The objective function and solution representation is given in the same section. For the GWO, three control parameters responsible for the search process should be initialized, which are C, A, and a. The first is responsible for exploration. The value range of this parameter takes a value of [0, 2]. The value of C is gradually updated and close to 2, when it is very close to the prey. Because this control parameter takes a random number during the course of run, it empowers the exploration features in the GWO [22]. The second and third parameters are interrelated (i.e. A and a) in which the first is calculated by the second as formulated in equation (16). It should be noted that the value range of the parameter A is $[-2, 2]$. The strength of exploration is raised when $A > 1$ or $A < -1$, while it is lowered when its value range is $-1 < A < 1$.

$$\vec{A} = 2\vec{a} \cdot \vec{r}_1 - \vec{a} \tag{10}$$

where $\vec{a}$ is a vector where its value range is linearly going down from 2 to 0 during the search. $\vec{r}_1$ are randomly generated vectors from the interval [0, 1]. The parameter a is updated by equation (11) given below:

$$a = 2 - t\left(\frac{2}{T}\right) \tag{11}$$

where t shows the current iteration, and T is the maximum number of iterations.

3.2 Initialize Population

The grey wolves' population memory ($\boldsymbol{X}$) is an augmented matrix of size $N \times d$. This memory contains the set of positions for the grey wolves (i.e. text document solutions) as many as N (see equation (12)). In order to generate any text document solution $\vec{X}$, the following formula is used:

$$X_i^j = U(1, 2, \ldots, K), \forall i = 1, 2, \ldots, n \text{ and } \forall j = 1, 2, \ldots, N$$

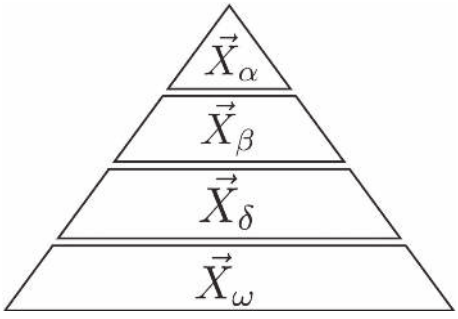

Figure 1: The Social Hierarchy for GWO Algorithm, Where the Number of Wolves are Decreased from Top Down ($\vec{X}_\alpha$, $\vec{X}_\beta$, $\vec{X}_\delta$, and $\vec{X}_\omega$), Respectively.

$U(1, 2, \ldots, n)$ generates a random digit in the range of 1 and K (K centroid). The objective function values $f(\vec{X})$ of each grey wolf position $\vec{X}$ are thereafter calculated by equation (8).

$$X = \begin{bmatrix} X_1^1 & X_2^1 & \cdots & X_n^1 \\ X_1^2 & X_2^2 & \cdots & X_n^2 \\ \vdots & \vdots & \cdots & \vdots \\ X_1^N & X_2^N & \cdots & X_n^N \end{bmatrix}. \tag{12}$$

3.3 Social Hierarchy

In the TCP-GWO, the social hierarchy of the natural grey wolf packs is adopted in the optimization rules in which the fittest three wolves in $\boldsymbol{X}$ are determined in greedy bases to hopefully turn the search toward the global optima. These solutions are $\vec{X}_\alpha$, $\vec{X}_\beta$, $\vec{X}_\delta$, which are the best, second-best, and third-best solutions, respectively. The remaining solutions are denoted as $\vec{X}_\omega$. Note that the group of wolves in $\vec{X}_\omega$ follows the locations of $\vec{X}_\alpha$, $\vec{X}_\beta$, $\vec{X}_\delta$ during the search [22, 37]. In text document clustering, the solution of the lowest fattiness function value is the best.

3.4 Encircling Prey

After the social hierarchy is adopted in the previous step, the intelligence behavior of the hunting process is also formulated in this step. According to the study conducted in Refs. [37] and [40], the hunting process naturally goes through three consecutive phases: (i) The prey is tracked, chased, and approached by the packs. (ii) The prey is encircled, harassed, and pursued to overstrain its effort. (iii) The prey is attacked by packs. These encircling phase is mathematically modeled as follows:

$$\vec{X}(t+1) = \vec{X}(t) - \vec{A} \cdot \vec{D} \tag{13}$$

where $\vec{X}(t+1)$ is the next location of any wolf, $\vec{X}(t)$ is the current location, $\vec{A}$ is a coefficient matrix, and $\vec{D}$ is a vector that is based on the location of the prey ($\vec{X}_p$), which is calculated as follows:

$$\vec{D} = \left| \vec{C} \cdot \vec{X}_p(t) - \vec{X}(t) \right| \tag{14}$$

where

$$\vec{C} = 2 \cdot \vec{r}_2.$$

Note that $\vec{r}_2$ are randomly generated vectors from the range [0,1]. With these two equations, a solution is able to relocate around another solution. Note that the equations use vectors, so this is applied to any number of dimensions, where $\vec{a}$ is a vector where its values are linearly decreased from 2 to 0 during the course of the run. $\vec{r}_1$ are randomly generated vectors from the interval [0,1]. The equation to update the parameter a is as follows:

$$a = 2 - t\left(\frac{2}{T}\right) \tag{15}$$

where t shows the current iteration, and T is the maximum number of iterations.

As mentioned above, the first step of hunting is to chase and encircle. To mathematically model this, the GWO considers two points in a d-dimensional space and updates the location of one of them based on that of another. The following equation has been proposed to simulate this.

Note that the random components in the above equations simulate different step sizes and movement speeds of grey wolves. The equation to define their values is as follows:

$$\vec{A} = 2\vec{a} \cdot \vec{r}_1 - \vec{a} \tag{16}$$

3.5 Search for Exploration

According to the above equations, a wolf can move to new points in a hyper-sphere around the target prey. However, this is insufficient to bridge the concept of the social hierarchy of the real and modeled GWO. As aforementioned, the social hierarchy is the key success of the hunting process. To achieve this, the founder of the GWO suggested to formulate the best three sets as α, β, and δ. In reality, each set has many solutions, though the founder considers one in each set. This is suggested to simplify the initial version of the GWO.

3.6 Attacking Prey

In the GWO, it is suggested that α, β, and δ are the best three solutions produced. The global optimal solution of the optimization problems is obviously unknown. Therefore, it is suggested that α, β, and δ have a very close position to the prey location. This is logical because they are the three best solutions in the entire population [25, 37]. Therefore, other wolves should be obliged to update their positions as follows:

$$\vec{X}(t+1) = \lfloor \frac{1}{3}\vec{X}_1 + \frac{1}{3}\vec{X}_2 + \frac{1}{3}\vec{X}_3 \rfloor \tag{17}$$

where $\vec{X}_1$ and $\vec{X}_2$ and $\vec{X}_3$ are calculated with equations (18), (19), and (20):

$$\vec{X}_1 = \vec{X}_\alpha(t) - \vec{A}_1 \cdot \vec{D}_\alpha \tag{18}$$
$$\vec{X}_2 = \vec{X}_\beta(t) - \vec{A}_2 \cdot \vec{D}_\beta \tag{19}$$
$$\vec{X}_3 = \vec{X}_\delta(t) - \vec{A}_3 \cdot \vec{D}_\delta \tag{20}$$

$\vec{D}_\alpha$, $\vec{D}_\beta$, and $\vec{D}_\delta$ are calculated using equations (21), (22), and (23):

$$\vec{D}_\alpha = \left| \vec{C}_1 \cdot \vec{X}_\alpha - \vec{X} \right| \tag{21}$$
$$\vec{D}_\beta = \left| \vec{C}_2 \cdot \vec{X}_\beta - \vec{X} \right| \tag{22}$$
$$\vec{D}_\delta = \left| \vec{C}_3 \cdot \vec{X}_\delta - \vec{X} \right| \tag{23}$$

3.7 Stop Criterion

The encircling and attacking prey steps are repeated until the maximum number of iterations T is reached. In checking for the stopping condition, it is required to check whether the *Lter* reaches the maximum value or not. If it reaches the maximum value, it is possible to get the best solution value. Otherwise, it is required to go to phase five.

The steps of text clustering using the proposed TCP-GWO are shown in Algorithm 1. Notably, the time complexity of the proposed GWO-based clustering is approximately equal to $\mathcal{O}(n \times K \times MaxIter \times N)$, where n represents the number of the document, K is the total number of clusters, *MaxIter* is the maximum number of iteration, and N is the number of solutions. Note that the complexity of the proposed GWO-based clustering method is calculated for each solution of the documents over the total amount of iterations. Particularly, the complexity of the objective function cost is $\mathcal{O}(n \times K)$.

4 Performance Criteria

The Precision $P(x, y)$, Recall $R(x, y)$, and F-score $F(x, y)$ metrics [13, 26] were considered to evaluate the performance of the text clustering results of a collection documents of class x in the cluster number y. These

Algorithm 1: A Pseudo Code of Text Clustering Problem using TCP-GWO.

MaxIter ← *Maximum number of iterations*
Lter ← *Iteration counter*
n ← *Population size (number of grey wolfs)*
G_i (Initial search agents) ($i = 1, 2, \ldots, n$)
Generate an initial population of GWO, as x_i ($i = 1, 2, \ldots, n$);
Define ϵ to be a predefined threshold value
Each wolf randomly selects n different document vectors, d, from the document database, D, and identifies them as the initial cluster centroid vectors.
Evaluate the initial population using the cost function defined in Equation 8
Initializing vector a, A and C
Estimation of the fitness value of each hunt agent;
  G_α = the best hunt agent
  G_β = the second best hunt agent
  G_δ = the third best hunt agent
Lter= 1
Repeat
for $i = 1 : G_s$ **do** //pack size of grey wolf
  Set each document vector, d, in the document database, D, to the nearest centroid vector, c_i.
  Evaluate the fitness value using Equation 8.
  Renew Current Hunt Agent Location
end for
Estimate fitness value for all grey wolfs using Equation 8
G_α, G_β and G_δ Updating
vectors a, A and C Updating
Lter=Lter+1
Until (Lter≥MaxIter is exceeded //stopping criteria OR the average change in centroid vectors (ADDC) $< \epsilon$)
Output G_s

measures were defined as shown in equations (24), (25), and (26):

$$P(x, y) = \frac{n_{xy}}{n_y} \tag{24}$$

$$R(x, y) = \frac{n_{xy}}{n_x} \tag{25}$$

$$F(x, y) = \frac{2 \times P(x, y) \times R(x, y)}{P(x, y) + R(x, y)} \tag{26}$$

where n_{xy} stands for the number of documents of class x in cluster y, n_y refers to the number of documents of cluster y, and n_x denotes the number of documents of class x for class x and cluster y.

The overall F-score measure is a result of the weighted average of $P(x, y)$ and $R(x, y)$ for each class x, as given in Equation (27).

$$F_a = \frac{\sum_x \left(|x| * F(x)\right)}{\sum_x |x|} \tag{27}$$

where $|x|$ is the size of class x.

For each pair of documents that share at least one cluster in the aggregation clustering results, these metric measures attempt to respect whether the prediction of this pair as in the same cluster was correct with regard to the underlying true categories of the data. Precision measure shows the proportion of the document pairs correctly put in the same cluster, recall relates to the proportion of the actual document pairs that were identified, and F-score is a weighted harmonic mean of precision and recall, which provides a comprehensive measure of performance of the TCP. It is used to find the proportion of truly distributed documents in each cluster and considered a standard criterion to assess the fineness of text clustering methods.

The accuracy (*Ac*) measure, as specified in equation (28) [1, 13], was also used to illustrate the performance of the proposed text document clustering approach. The *Ac* criterion, as an external measure, was originally defined as the percentage of truly assigned documents to the clusters over all document collections.

$$Ac = \frac{1}{n}\sum_{i=1}^{K} P(x, y) \tag{28}$$

where n and K represent the number of documents per cluster and the number of all clusters, respectively.

5 Experimental Results

This section describes the experimental study to verify the effectiveness of the proposed TCP-GWO method. Specifically, it presents the text clustering performance in terms of the aforesaid evaluation criteria, statistical analysis, and a comparison of the results with other TCP-based text clustering approaches.

Two other text clustering approaches were presented here to comparatively evaluate our proposed approach. These approaches are the TCP-based hill climbing technique by Abualigah et al. [4] and the TCP-based β-hill climbing technique [23]. The parameters used for all the approaches including the GWO are as follows: the number of experiments is 20 times with each run embraced of 1000 iterations to fit with the experimental setting reported by the authors. These TCP-based text clustering approaches were implemented in Matlab 7.10 (R2010a). All the experiments are run using Dell computers with a Windows 7 of 64-bit professional and 64 GB of RAM.

5.1 Text Document Data Sets

To evaluate the proposed TCP-GWO, we assessed it on six text documents selected randomly from available public data sets published by Dmoz-Business data set. The size and complexity issues of these data sets vary depending on the selection mechanism. These selected data sets are described in Table 2. Table 2 shows the number of documents, terms, and clusters of each data set.

The first data set, DS1, consisting of 200 documents, is represented by 5773 terms and pertains to 10 clusters. The second data set, DS2, possesses 299 documents, represented by 1725 terms where it belongs to a category of four clusters. The third data set, DS3, of 100 documents, is represented by 3236 terms and comports to five cluster classes. The fourth data set, DS4, contains 333 documents, represented by 4339 terms. This set belongs to a category of four clusters. The fifth data set, DS5, contains 1660 documents, represented by 8659 terms. This set belongs to a class of five clusters. Finally, the sixth data set, DS6, contains 2301 documents, represented by 12,942 terms and is connected to six cluster categories.

5.2 Results and Discussions

The performance of the proposed TCP-GWO in terms of the evaluation criteria is shown for up to 1000 iterations in Figure 2.

Table 2: Description of the Selected Text Documents (Data Sets) Used.

Dataset	# of Documents	# of Terms	# of Clusters
DS1	200	5773	10
DS2	299	1725	4
DS3	100	3236	5
DS4	333	4339	4
DS5	1660	8659	5
DS6	2301	12,942	6

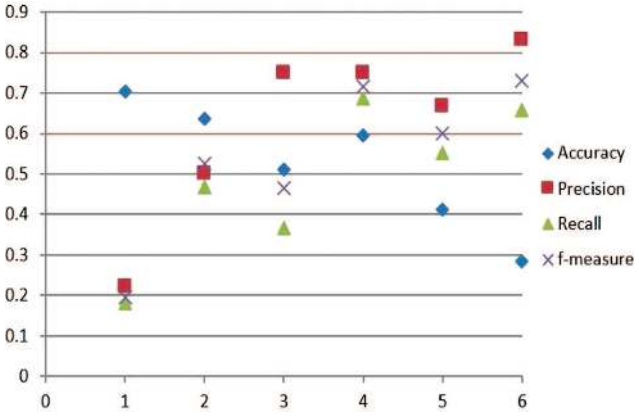

Figure 2: The Final TCP-GWO Responses to Clustering Documents.

Figure 2 shows the clustering results of the TCP-GWO in each data set with convincing accuracy rate. In the first three cases, relatively high accuracy and recall rates were achieved. This proves that the GWO-based swarm optimization algorithm succeeded in escaping from the local optima to reach the optimal solution. The TCP-GWO reported, respectively, 0.7037, 0.2222, 0.1794, and 0.1957 for accuracy, precision, recall, and F-measure for the documents selected from DS1. Further, it reported performance values of 0.6349, 0.5000, 0.4677, 0.5249, 0.5113, 0.7500, 0.3660, 0.4645, 0.5944, 0.7500, 0.6873, 0.7156, 0.4143, 0.6667, 0.5542, 0.5990, 0.2840, 0.8333, 0.6598, and 0.7305 for the accuracy, precision, recall, and F-measure rates for the documents selected from DS2, DS3, DS4, DS5, and DS6, respectively. The evaluation results in Figure 2 show the appropriateness of the proposed text clustering method for the tested documents. It is clear that a relatively high level of performance was achieved using the proposed TCP-GWO method for DS1 and DS2. This illustrates that the TCP-GWO method was well-managed to be an adept text clustering system for the selected data sets. This implies that the GWO was well adapted to generate a thoroughly stable clustering approach toward the end of the clustering process. However, the accuracy of the clustering results of the TCP-GWO method for the first three data sets is considerably better than the accuracy of the last three data sets. This is considered a big limitation of the TCP-GWO method. The last three data sets may particularly contain documents of extremely large sizes or address documents of variable complexity, as the TCP-GWO approach did not behave well with the high-complexity documents that are present in the selected data sets. This partly hindered the reliability of the developed approach.

The evaluation results of the TCP-GWO for the selected document data sets are also summarized in Table 3 in terms of the evaluation measures (i.e. accuracy, precision, recall, and F-measure).

It is observed in Table 4 that the recorded results of the TCP-GWO are at a high degree of performance, such that the computed accuracy results are highly reasonable.

The performance of the proposed TCP-GWO and the other two evaluated text clustering methods in the clustering of the six randomly selected data sets is presented in Table 3.

Table 3: The Evaluation Results of the TCP-GWO Arrived up to 20 Experiments on the Clustered Data Set.

Data sets	Accuracy	Precision	Recall	F-measure
D1	0.7037	0.2222	0.1794	0.1957
D2	0.6349	0.5000	0.4677	0.5249
D3	0.5113	0.7500	0.3660	0.4645
D4	0.5944	0.7500	0.6873	0.7156
D5	0.4143	0.6667	0.5542	0.5990
D6	0.2840	0.8333	0.6598	0.7305

Table 4: A Comparison Performance Between TCP-based GWO, Hill Climbing, and β-hill Climbing in terms of the Reported Best Performance Text Clustering Results.

Data sets	Evaluation measures *Ac*, *P*, *R*, and *F*	Hill climbing	β-Hill climbing	GWO
D1	Accuracy	0.1900	0.2150	0.7037
	Precision	0.1875	0.2032	0.2222
	Recall	0.2024	0.2188	0.1794
	F-measure	0.1946	0.2107	0.1957
D2	Accuracy	0.2853	0.3093	0.6349
	Precision	0.2850	0.3056	0.5000
	Recall	0.2847	0.3090	0.4677
	F-measure	0.2849	0.3073	0.5249
D3	Accuracy	0.2994	0.3144	0.5113
	Precision	0.2899	0.3149	0.7500
	Recall	0.2871	0.3203	0.3660
	F-measure	0.2885	0.3176	0.4645
D4	Accuracy	0.3482	0.3144	0.5944
	Precision	0.3476	0.3149	0.7500
	Recall	0.3487	0.3203	0.6873
	F-measure	0.3482	0.3176	0.7156
D5	Accuracy	0.3392	0.3536	0.4143
	Precision	0.3382	0.3534	0.6667
	Recall	0.3365	0.3558	0.5542
	F-measure	0.3374	0.3546	0.5990
D6	Accuracy	0.2673	0.3123	0.2840
	Precision	0.2615	0.3195	0.8333
	Recall	0.2572	0.3198	0.6598
	F-measure	0.2593	0.3196	0.7305

The TCP-GWO's results in Table 4 demonstrate a highly convincing level of clustering for the first three data sets. It is also clear that the TCP-GWO achieved a faintly better degree of performance than hill climbing and β-hill climbing-based clustering schemes. The differences are remarkably significant between the TCP-GWO and TCP-β-hill climbing and TCP-hill climbing methods, which are, on average, about 0.2206 and 0.23553, respectively. Also, it is observed in Table 4 that the TCP-based β-hill climbing method achieved better results to those reported by the TCP-based-hill climbing method.

As Table 4 illustrates, the proposed TCP-GWO method shows superior performance compared to the other text clustering approaches. It achieved the highest accuracy and recall values among all the evaluated methods. This is related to the improvement realized by the GWO as an optimization algorithm for the clustering scheme.

Table 4 shows that the proposed GWO-TCP overcomes other comparative methods almost in the most experimented cases. According to the accuracy value, the GWO-TCP excels in the high-ranked method by getting the best results in five out of six text document data sets (i.e. DS1, DS2, DS3, DS4, and DS5). The β-hill climbing technique got the best result in one out of six text document data sets (i.e. DS6). According to the precision values, the GWO-TCP can be flagged as a high-rank method by getting the best results on all the text document data sets for all experimented cases (i.e. DS1, DS2, DS3, DS4, DS5, and DS6). According to the recall value, the GWO-TCP reaches the high-rank method by getting the best results in five out of six data sets (i.e. DS2, DS3, DS4, DS5, and DS6). The β-hill climbing technique achieved one best result out of six (i.e. DS1).

To gain a deeper insight into the clustering results and to verify the efficiency of the TCP-GWO method, a statistical test analysis was conducted. This is to rank each method along with the others and to identify the best text clustering approach. The statistical results on the basis of the F-measure value are presented in Table 5.

In Table 5, the TCP-GWO method is ranked first, followed by the TCP-β-hill climbing, and the last one is the TCP-hill climbing by a relatively large margin. The results shown in Table 5 confirm that the TCP-GWO

Table 5: Statistical analysis based on F-measure values.

Dataset	Hill climbing	β-Hill climbing	GWO
DS1	3	1	2
DS2	3	2	1
DS3	3	2	1
DS4	2	3	1
DS5	3	2	1
DS6	3	2	1
Mean rank	2.83	2.00	1.16
Final rank	3	2	1

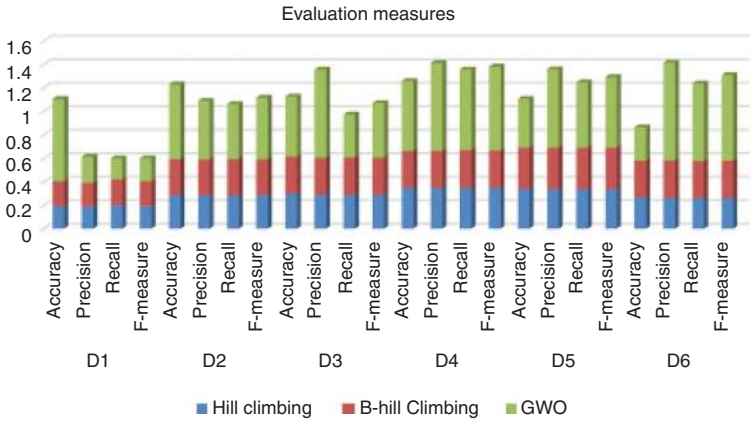

Figure 3: Comparison Between TCP-GWO and Other Comparative Approaches.

method, statistically, exhibited a respectable level of performance, much better than the TCP-hill climbing and TCP-β-hill climbing.

Table 5 shows the statistical analysis according to the F-measure values. It is clear that the GWO achieved the best ranking (i.e. 1) according to the statistical analysis with regard to the common evaluation measures used in the domain of the text clustering. The second best was the β-hill climbing; it got the second best ranking (i.e. 2). Finally, the worst method was hill climbing; it achieved the third ranking (i.e. 3).

Finally, according to the F-measure value, the TCP-GWO achieved the high-rank method by getting the best results in five out of six data sets (i.e. DS2, DS3, DS4, DS5, and DS6). The β-hill climbing technique achieved one of the best results out of six (i.e. DS1). We observed from Figure 3 that the performance of the GWO for solving the text clustering problem was very well in comparison with the state-of-the-art methods in terms of all the evaluation measures.

6 Conclusion and Future Work

The work presented demonstrated the use of the GWO as a metaheuristic swarm-based algorithm for solving the TCP, referred to as the TCP-GWO. The aim is an efficient and beneficial scheme of text clustering. We have used the ADDC as a fitness function of the GWO to accomplish the text clustering approach for six text documents selected from a public text document benchmarking. The proposed approach introduced a new method of text clustering by exploiting the GWO. This method has a good level of reliability. The allocation of documents to the best clusters was improved by GWO, which also helps to flee from local optimum solutions to the best global solutions. A set of evaluation measures was employed to evaluate the performance of the TCP-GWO.

The flexibility of the TCP-GWO scheme in the reliable clustering text documents is shown for the documents selected from six data sets with documents of various sizes and scales with simple and complex

texts and for documents with variable numbers of terms and high levels of complexity. The precision and recall rates show a high degree of sensitivity and specificity. On all criteria, a very high level of performance is shown. The evaluation results showed that the proposed TCP-GWO revealed an improvement in the performance criteria of accuracy of more than 9% of the clustering results on the six data sets over the other comparative clustering methods. The TCP-GWO approach is a potentially promising new method of text clustering that merits further study.

As can be borne out by the results, the proposed TCP-GWO is able to produce results better than those produced by the β-hill climbing and hill climbing algorithms for all the test data sets. The maneuver and strong features of the TCP-GWO, represented by its ability to share the useful knowledge from the α, β, and δ solutions to other members, enable it to navigate the TCP search space in a very effective manner. Furthermore, the GWO operators are able to strike the right trade-off between the local nearby exploitation and wider exploration of search concepts, thus, enriching the convergence features. It is worth mentioning that the exploration is the ability of the algorithm to reach unvisited search space regions, while exploitation refers to the ability to make use of the accumulative search using the current objective function. In a nutshell, the TCP-GWO is a very efficient search strategy for clustering-orientation problems that can be widely used for the same purpose, and it is pregnant with a tremendous development in the future. There are many trends for future work that could be considered:

- The comparison of alternative clustering problems with various levels of complexity of the documents.
- The evaluation of adaptability to different data sets and terms.
- Increasing the robustness and accuracy of TCP using different metaheuristic approaches as well as other fitness functions.

Bibliography

[1] L. M. Abualigah, A. T. Khader and M. A. Al-Betar, 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, Amman, Jordan, 2016.

[2] L. M. Abualigah, A. T. Khader and M. A. Al-Betar, Unsupervised feature selection technique based on genetic algorithm for improving the text clustering, in: *Computer Science and Information Technology (CSIT), 2016 7th International Conference on*, IEEE, pp. 1–6, Amman, Jordan, 2016.

[3] L. M. Abualigah, A. T. Khader, M. A. Al-Betar and M. A. Awadallah, A krill herd algorithm for efficient text documents clustering, in:*Computer Applications and Industrial Electronics (ISCAIE), 2016 IEEE Symposium on*, IEEE, pp. 67–72, 2016.

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

[5] L. M. Abualigah, A. M. Sawaie, A. T. Khader, H. Rashaideh, M. A. Al-Betar and M. Shehab, β-hill climbing technique for the text document clustering, *New Trends Inf. Technol.* **60** (2017), 60–66.

[6] A. Agarwal, A. Chandra, S. Shalivahan and R. K. Singh, Grey wolf optimizer: a new strategy to invert geophysical data sets, *Geophys. Prospect.* **66** (2018), 1215–1226.

[7] M. A. Al-Betar and M. A. Awadallah, Island bat algorithm for optimization, *Exp. Syst. Appl.* **107** (2018), 126–145.

[8] M. A. Al-Betar, M. A. Awadallah, H. Faris, X.-S. Yang, A. T. Khader and O. A. Alomari, Bat-inspired algorithms with natural selection mechanisms for global optimization, *Neurocomputing* **273** (2018), 448–465.

[9] Z. A. Al-Sai and L. M. Abualigah, Big data and e-government: a review, in: *Information Technology (ICIT), 2017 8th International Conference on*, IEEE, pp. 580–587, Amman, Jordan, 2017.

[10] Z. A. A. Alyasseri, A. T. Khader, M. A. Al-Betar, M. A. Awadallah and X.-S. Yang, Variants of the flower pollination algorithm: a review, in: *Nature-Inspired Algorithms and Applied Optimization*, pp. 91–118, Springer, Cham, 2018.

[11] M. A. Awadallah, M. A. Al-Betar, A. L. Bolaji, E. M. Alsukhni and H. Al-Zoubi, Natural selection methods for artificial bee colony with new versions of onlooker bee, *Soft Comput.* **22** (2018), 1–40.

[12] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*, Vol. 463, ACM Press, New York, 1999.

[13] M. W. Berry and M. Castellanos, Survey of text mining ii: clustering. Classification, and retrieval 1, 2007.

[14] K. K. Bharti and P. K. Singh, Chaotic gradient artificial bee colony for text clustering. *Soft Comput.* **20** (2016), 1113–1126.

[15] A. L. Bolaji, M. A. Al-Betar, M. A. Awadallah, A. T. Khader and L. M. Abualigah, A comprehensive review: Krill herd algorithm (kh) and its applications, *Appl. Soft Comput.* **49** (2016), 437–446.

[16] V. Chahar, J. Chhabra and D. Kumar, Grey wolf algorithm-based clustering technique, *Journal of Intelligent Systems* **26** (2016), 153–168.

[17] O. Chum, J. Philbin and A. Zisserman, Near duplicate image detection: min-Hash and TF-IDF weighting, *BMVC* **810** (2008), 812–815.

[18] K J. Cios, W. Pedrycz and R. W. Swiniarski, Rough sets, in: *Data Mining Methods for Knowledge Discovery*, pp. 27–71, Springer, Boston, MA, 1998.

[19] X. Cui, T. E. Potok and P. Palathingal, Document clustering using particle swarm optimization, in: *Swarm Intelligence Symposium, 2005, SIS 2005, Proceedings 2005 IEEE*, IEEE, pp. 185–191, Pasadena, CA, USA, 2005.

[20] T.-K. Dao, Enhanced diversity herds grey wolf optimizer for optimal area coverage in wireless sensor networks, in: *Genetic and Evolutionary Computing: Proceedings of the Tenth International Conference on Genetic and Evolutionary Computing*, November 7–9, 2016 Fuzhou City, Fujian Province, China, Vol. 536, Springer, p. 174, 2016.

[21] E. Emary, H. M. Zawbaa, C. Grosan and A. E. Hassenian, *Feature Subset Selection Approach by Gray-wolf Optimization*, Springer International Publishing, Cham, pp. 1–13, 2015.

[22] H. Faris, I. Aljarah, M. A. Al-Betar and S. Mirjalili, Grey wolf optimizer: a review of recent variants and applications, *Neural Comput. Appl.* **30** (2017), 413–435.

[23] R. Forsati, M. Meybodi, M. Mahdavi and A. Neiat, Hybridization of k-means and harmony search methods for web page clustering, in: *Web Intelligence and Intelligent Agent Technology, 2008. WI-IAT'08. IEEE/WIC/ACM International Conference on*, Vol. 1, IEEE, pp. 329–335, Sydney, Australia, 2008.

[24] A.-S. Ghadeer, I. Aljarah and H. Alsawalqah, Enhancing the Arabic sentiment analysis using different preprocessing operators, *New Trends Inf. Technol.* **113** (2017), 113–117.

[25] S. Gupta and K. Deep, A novel random walk grey wolf optimizer, *Swarm Evol. Comput.* (2018).

[26] T. Hu and S. Y. Sung, Finding centroid clusterings with entropy-based criteria, *Knowl. Inf. Syst.* **10** (2006), 505–514.

[27] A. K. Jain, M. N. Murty and P. J. Flynn, Data clustering: a review, *ACM Comput. Surv. (CSUR)* **31** (1999), 264–323.

[28] J. Jayapriya and M. Arock, Aligning two molecular sequences using genetic operators in grey wolf optimiser technique, *Int. J. Data Min. Bioinform.* **15** (2016), 328–349.

[29] K. Kanimozhi and M. Venkatesan, A novel map-reduce based augmented clustering algorithm for big text datasets, in: *Data Engineering and Intelligent Computing*, pp. 427–436, Springer, Berlin, Heidelberg, Germany, 2018.

[30] J. Kennedy and Y. Shi, *Swarm Intelligence. The Morgan Kaufmann Series in Evolutionary Computation*, Elsevier Science & Technology, Elsevier, Amsterdam, The Netherlands, 2001.

[31] N. Kushwaha and M. Pant, Link based BPSO for feature selection in big data text clustering, *Future Gener. Comput. Syst.* **82** (2017), 190–199.

[32] D. K. Lal, A. Barisal and M. Tripathy, Grey wolf optimizer algorithm based fuzzy PID controller for AGC of multi-area power system with TCPS, *Procedia Comput. Sci.* **92** (2016), 99–105.

[33] C. Lu, L. Gao, X. Li and S. Xiao, A hybrid multi-objective grey wolf optimizer for dynamic scheduling in a real-world welding industry, *Eng. Appl. Artif. Intell.* **57** (2017), 61–79.

[34] S. Medjahed, T. A, Saadi, A. Benyettou and M. Ouali, Gray wolf optimizer for hyperspectral band selection, *Appl. Soft Comput.* **40** (2016), 178–186.

[35] D. Merkl, Industry: text mining with self-organizing maps, in: *Handbook of Data Mining and Knowledge Discovery*, pp. 903–910, Oxford University Press, Inc., New York, NY, USA, 2002.

[36] S. Mirjalili, How effective is the grey wolf optimizer in training multi-layer perceptrons, *Appl. Intell.* **43** (2015), 150–161.

[37] S. Mirjalili, S. M. Mirjalili and A. Lewis, Grey wolf optimizer, *Adv. Eng. Softw.* **69** (2014), 46–61.

[38] M. Mosavi, M. Khishe and A. Ghamgosar, Classification of sonar data set using neural network trained by gray wolf optimization, *Neural Netw. World* **26** (2016), 393.

[39] A. Mostafa, Fouad, M. Houseni, N. Allam, A. E. Hassanien, H. Hefny and I. Aslanishvili, A hybrid grey wolf based segmentation with statistical image for ct liver images, in: *International Conference on Advanced Intelligent Systems and Informatics*, pp. 846–855, Springer, Berlin, Heidelberg, Germany, 2016.

[40] L. K. Panwar, S. Reddy, A. Verma, B. Panigrahi and R. Kumar, Binary grey wolf optimizer for large scale unit commitment problem, *Swarm Evol. Comput.* **38** (2018), 251–266.

[41] M. H. Qais, H. M. Hasanien and S. Alghuwainem, Augmented grey wolf optimizer for grid-connected PMSG-based wind energy conversion systems, *Appl. Soft Comput.* (2018).

[42] V. V. Raghavan and K. Birchard, A clustering strategy based on a formalism of the reproductive process in natural systems, in: *ACM SIGIR Forum*, Vol. 14, pp. 10–22, ACM, New York, NY, 1979.

[43] R. A. Saravanan and M. R. Babu, Enhanced text mining approach based on ontology for clustering research project selection, *J. Ambient Intell. Humaniz. Comput.* (2017), 1–11. DOI: 10.1007/s12652-017-0637-7.

[44] X. Song, L. Tang, S. Zhao, X. Zhang, L. Li, J. Huang and W. Cai, Grey wolf optimizer for parameter estimation in surface waves, *Soil Dyn. Earthq. Eng.* **75** (2015), 147–157.

[45] H. C. Tijms, *Stochastic Models: An Algorithmic Approach*, Vol. 303, John Wiley & Sons Inc, Hoboken, NJ, USA, 1994.

[46] M. M. Zaw and E. E. Mon, Web document clustering by using PSO-based cuckoo search clustering algorithm, in: *Recent Advances in Swarm Intelligence and Evolutionary Computation*, pp. 263–281, Springer, Berlin, Heidelberg, Germany, 2015.
[47] S. Zhang and Y. Zhou, Grey wolf optimizer based on Powell local optimization method for clustering analysis, *Discrete Dyn. Nat. Soc.* **2015** (2015), Article ID 481360, 17 pages. http://dx.doi.org/10.1155/2015/481360.
[48] S. Zhang and Y. Zhou, Template matching using grey wolf optimizer with lateral inhibition, *Optik* **130** (2017), 1229–1243.
[49] S. Zhang, Y. Zhou, Z. Li and W. Pan, Grey wolf optimizer for unmanned combat aerial vehicle path planning, *Adv. Eng. Softw.* **99** (2016), 121–136.
[50] Y. Zhao and G. Karypis, Empirical and theoretical comparisons of selected criterion functions for document clustering, *Mach. Learn.* **55** (2004), 311–331.