Journal of Autonomous Intelligence (2024) Volume 7 Issue 5
doi: 10.32629/jai.v7i5.1631

ORIGINAL RESEARCH ARTICLE

Enhanced feature selection with bacterial foraging and rough set analysis for document clustering

S. Periyasamy[1,*], R. Kaniezhil[2]

[1] *Department of Computer Science, Periyar University, Salem, Tamilnadu 636 011, India*

[2] *Navarasam College of Arts and Science for Women, Erode, Tamilnadu 638 101, India*

*** Corresponding author:** S. Periyasamy, speriyasamyphd@yahoo.com

ABSTRACT

Most applications, such as Information Retrieval and Natural Language Processing (NLP), utilize document clustering to improve their analysis. The document consists of various features that are utilized to determine the similar and dissimilar documents. However, the traditional techniques consume high computation difficulties and convergence problems while analyzing high-dimensional data. The research difficulties are addressed with the help of Bacterial Foraging and Rough Set Analysis (BF-RSA). This study uses the TF-IDF features for analyzing similar documents. The extracted features are explored using the Bacterial Foraging Optimization (BFO) approach that uses the exploration and exploitation characteristics to improve the overall clustering quality. The collected documents are analyzed using a roughest approach that generates the discernible matrix which helps to identify similar and dissimilar features. Then bacterial foraging method computes the fitness value according to their behavior to identify the optimal solution. The selected feature set is further analyzed in the roughest approximation condition to minimize the uncertainty and interpretability issues. The effective integration of bacteria foraging and rough set approach maximizes the feature selection accuracy and improves the clustering accuracy (97.05%) with minimum convergence speed (0.063 s).

Keywords: document clustering; information retrieval; bacterial foraging and rough set analysis (BF-RSA); uncertainty and interpretability

ARTICLE INFO

Received: 5 March 2024
Accepted: 9 April 2024
Available online: 30 May 2024

1. Introduction

Document Clustering (DC)[1–3] is one of the centralized processes in which descriptors are extracted to improve the text analysis. The DC is utilized in both online and offline applications to perform similar document clustering, feedback analysis, corpus summarization, information filtering, topic extraction, and document classification[4]. Document clustering is explored to maximize the recall value of information retrieval systems. The clustering process automatically creates documents with hierarchical clusters[5]. Different clustering approaches are utilized to explore the text taxonomy which helps to generate the effective document classification. The documents consist of various comments, sentences, words, posts, reviews, articles, tweets, books, papers, web pages, etc.[6]. Similar documents are explored and grouped with the help of Natural Language Processing (NLP) and clustering techniques[7]. The grouping is done according to similar features and content, which helps to attain the clustering goal. Text clustering[8] intends to explore the document structure to analyze, manage, and understand the data. Document clustering has several steps, such as

text representations, feature extraction, clustering, and evaluation.

The text representation uses the labelled sentence data in which every document is represented as a numerical vector (text vectorization). In text vectorization, textual information is converted into a suitable processing format. During the analysis, Term Frequency (TF)[9,10] and Inverse Term Frequency (ITF) are utilized to derive extract vectors from the text. The relevant features should be extracted from the text which includes the n-gram frequency[11], word frequencies, and linguistic elements[12]. The feature extraction procedure is selected depending on the clustering requirement. The derived feature is fed into the clustering algorithms[13–15] like K-means, DBSCAN, hierarchical clustering approaches, etc. The clustering process groups similar features, which helps to improve pattern recognition, text summarization, topic modeling, and content recommendations[16]. However, document clustering has several challenges while handling high-dimensional data because documents are represented as high-dimensional vectors. Every document related to unique terms in the document leads to the dimensionality issues that are directly linked with the computational expensive problems. The document matrices are generally spars, which means document entries are zero. The sparsity problem[17] in clustering causes unreliable similarity among the documents. Most of the time, documents have polysemy with synonymy, which causes ambiguity, which is one of the challenging tasks while clustering. In addition, the small-scale patterns are not relevant to the large contexts, and the scaling issues influence the cluster quality[18]. Therefore, the validation and optimal number of clusters are difficult to determine.

The research issues are addressed by including the feature selection steps in document clustering. The feature selection reduces the dimensionality issues by selecting the more relevant features. The selected features mitigate the dimensionality curse and improve the overall clustering efficiency. The selected document features highly discriminative to the terms and concentrates only on relevant information. The selection process excludes the noise and irrelevant, confusion, and negative features which directly linked with the interpretability and quality of clusters. The reduced and optimal set of features based on formed clusters is more interpretable and understandable. Finally, the feature selection procedure improves computational efficiency, even when analyzing large document collections. Several feature selection techniques, like genetic algorithms, wrapper techniques, filtering approaches, and optimization algorithms, are widely applied to the feature selection process. However, these methods are difficult to address the above discussed issues such as high dimensionality and cluster quality issues.

The research difficulties are addressed with the help of Bacterial Foraging and Rough Set Analysis (BF-RSA). The bacterial foraging optimization approaches balance the exploitation and exploration process which helps to select the optimal features. The optimized feature selection procedure minimizes the high-dimensionality issue. The selected discriminative features understand the feature balancing between the combination of features. The relevant features are changed in the dynamic environment that affects the cluster quality. The bacterial foraging method's adaptability characteristics help to select the relevant features from the collection of documents. In addition, the optimization algorithm created for global optimization improves the clustering quality. The rough set approach addresses the vagueness and uncertainty issues by selecting robust features. The rough set analyzes document features that produce interpretable results and understand the document features effectively. The rough set approach allows the information granulations that identify the relationship between the patterns and discernibility matrics, which reduces the computation complexity. Thus, the research utilizes the rough set along with the bacterial foraging optimization methods to solve the issues in the dynamic environment, high dimensionality, interpretability, and uncertainty issues in document clustering. The effective utilization of optimization procedure and analytical process helps to predict the subset features which helps to maximize the clustering accuracy and quality. The objective of this research is defined as follows.

- To reduce the dimensionality issues by utilizing the bacterial foraging optimization adaptability while selecting the subset features.
- To minimize the uncertainty problem, incorporate the rough set theory to identify the relationship between the document features.
- To improve the cluster quality and accuracy by identifying the similarity between the texts.

Then, the rest of the paper is prearranged as follows. Section 2 discusses the various researcher's opinions regarding document clustering. Section 3 describes the working process of Bacterial Foraging and Rough Set Analysis (BF-RSA) based document clustering, and the efficiency of the system is evaluated in section 4. Conclusion described in section 5.

2. Related work

Abualigah et al.[19] introduced the particle swarm optimization (PSO) method to maximize the document clustering process. This study intends to address the optimization problems while handling irrelevant, unnecessary, and noisy features. The collected information is processed to select the features, which is done by applying the particle swarm method. The feature selection process minimizes the computational time while selecting the features. The selected features are fed into the text clustering method that groups similar features. This study uses the term frequency and inverse term frequencies to derive the features at various levels of the document. The effective derivation features maximize the overall document clustering efficiency.

Christy and Gandhi[20] applied a random feature set generation approach (RFSG) to improve the feature selection and document clustering process. This study uses the filtering-based feature selection approach that extracts the features according to the quality metric. The derived features are processed using a random feature set generation approach. The features are ranked, and the best features are selected and clustered using K-means clustering and X-means algorithm. The method achieves a 0.75 R square value while clustering a large dataset.

Lakshmi and Baskar[21] introduced Dissimilarity Document K-Means Clustering (DIC-DOC-K-means) for improving text document clustering. Initially, the cluster centroid is selected by checking the standard deviation value of term frequency. If the feature has the minimum standard deviation values, it is selected as the centroid. Then subsequent centroid is choosen according to the dissimilarities of the previously chosen centroid. According to the distance measure, the similar features are clustered by solving the class invariance problem. The DIC-DOC-K means method forms the cluster with different sizes, such as 4, 8, 12, and 16, and attains maximum clustering accuracy.

Bezdan et al.[22] recommended the Hybrid Fruit Fly Optimized K-means (HFFO-K-Means) approach to perform the document clustering. The research intends to address the complexity involved in unstructured data analysis of large volumes of data. The hybridized approach explores every data and addresses the difficulties while partitioning the large dataset. The cluster center is selected based on the swarm intelligence approach, and the selected features are grouped depending on the k-means algorithm function. The discussed optimization algorithm works on the CEC2019 benchmark function in which the system ensures high clustering accuracy by solving the computation complexity.

Kim et al.[23] maximized sparse centroid projection and initialization using the Spherical K-means clustering (SK-means). The system aims to address the convergence issues while clustering high-dimensional data. The centroid vector sparsity value is computed using the data-driven threshold value. The selected vectors are further explored using an unsupervised clustering labeling approach, which derives the features from the text. Finally, K-means clustering is applied to form the cluster, which resolves the convergence issues effectively.

Wang et al.[24] applied Parallelized K-Means Clustering (PK-Means) approach to improve the text clustering efficiency. The system needs to solve the poor calculation efficiency and high dimensionality issues. The research difficulties are addressed using the k-means clustering, spark big data, and Hadoop approach. Initially, word2vector is applied to compute the word vector weight values that minimize the dimensionality issues. Then canopy approach is applied to the data weight value to predict the cluster center. After that k-means clustering approach is incorporated into preprocessed data to improve clustering efficiency.

Abualigah et al.[25] roposed Nature inspirted optimized approach (NIOA) for improving the document clustering accuracy. This study explores various optimization algorithms such as Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Harmony Search (HS), Cuckoo Search algorithm (CSA), Bat-Inspired approach (BIA), Gray Wolf Optimization (GWO), and Krill Herd Algorithm (KHA) to improve the overall text clustering. The optimization algorithms are used to address the optimization problems while exploring data features.

Cekik and Uysal[26] performed text clustering using a filter rough set feature selection approach. The documents are examined frequently, and terms are extracted using a rough set. Then, term vector sparsity information is estimated using the same rough set. The extracted terms distances are computed, and similar features are grouped, which improves the text analysis.

Abualigah et al.[27] introduced a Meta-heuristic Optimization Algorithm (MHOA) for improving text clustering. This study intends to resolve the clustering problems by applying different optimization techniques such as genetic algorithm, social spider optimization, harmony search, particle swarm algorithm, and hybridized approach. These approaches give the solution while selecting the cluster centroid that maximizes the overall text clustering.

Ibrahim et al.[28] recommended semantic similarity computations to perform document clustering. The system computes the meaning of the documents using semantic measures and similar documents are clustered. During the analysis, 50 research papers were selected, of which 13 researchers works are selected which are relevant to the semantic similarity. Then, deep analysis was conducted to identify the effective methods for improving text clustering.

Chen et al.[29] introduced a hybrid optimized approach for feature selection. This study combines the particle swarm algorithm and spiral-shaped mechanism to select the optimal features. The feature selection approach works according to the wrapper approach, which reduces the computation difficulties. Initially, a logistic map sequence is applied to maximize the diversity in searching. The new generation position is updated according to the spiral-shaped approach that minimizes the local search problem and provides optimal solutions.

Abasi et al.[30] applied the Link-Based Multi-Sever Optimization Technique (LMSO) to improve document clustering. This study intends to address the text document clustering problem and maximize learning efficiency. The optimization algorithm selects the best solution with a minimum low convergence rate and local optima solution. During the searching process, a neighborhood selection technique is applied that computes the probability factor for every feature. The selected features fed into the clustering process for maximizing the clustering accuracy. According to the various researcher's opinions, the meta-heuristic optimization algorithm-based feature selection process is widely utilized to improve text or document clustering. However, the system consumes high computation complexity while handling large volumes of data. In addition, similar feature identification requires the learning technique to improve the overall clustering techniques. Therefore, this study uses the optimized rough set approach for selecting document features. The detailed working process of feature selection is described in the below section.

3. Process of document clustering

The research focuses on improving the clustering quality accuracy and minimizing the high-dimensionality issues during large volumes of data analysis. During the analysis, document clustering faces uncertainty, interpretability, and robust issues. The research issues are addressed by embedding the feature selection process in the document clustering process. Here, document text representation is explored in terms of words and sentence-related vectors. Then, features are derived from the texts which may irrelevance to the clustering and create high computation time. Therefore, the feature selection step is included by using Bacterial Foraging and Rough Set Analysis (BF-RSA) to select the appropriate and optimized features. The bacterial foraging optimization process has adaptability characteristics that resolve the high-dimensionality issues. The roughest approach addresses the uncertainty and interpretability issues during large data analysis. The BF-based selected features maximize the clustering accuracy and quality. The overall working process of the BF-RSA feature selection process is illustrated in **Figure 1**.

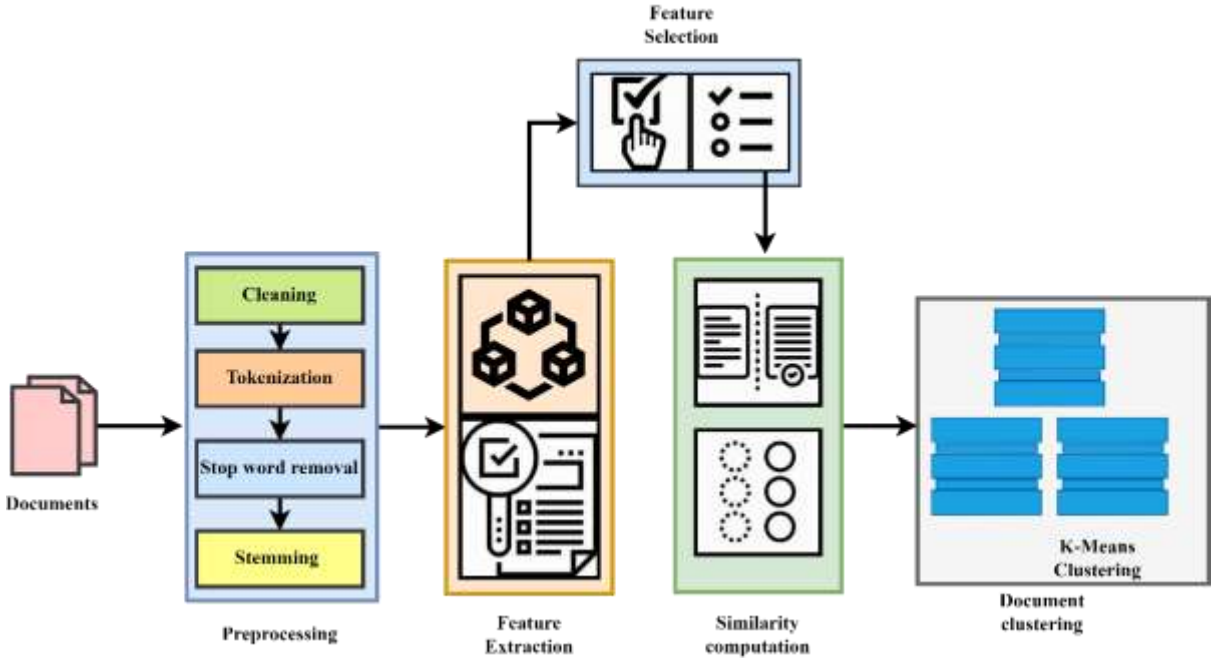


Figure 1. Structure of feature selection based document clustering.

Figure 1 demonstrates the document clustering process, which includes several steps such as document collection, noise removal process, feature extraction, selection, and clustering. Here, Bacterial Foraging and Rough Set Analysis (BF-RSA) are utilized for the feature selection that selects the optimized features, which helps to maximize the clustering quality and accuracy. This study uses the BBC dataset information[31] that is collected from BBC News, which consists of non-commercial and research analysis information. The dataset has 2225 documents gathered from 2004 to 2005 year that include various field information like sports, business, entertainment, politics, and tech. The collected texts consist of irrelevant and inconsistent information, which reduces the clustering efficiency. Therefore, data inconsistency is reduced by applying cleaning, stopword removal, tokenization, and stemming. The detailed noise removal process is illustrated in **Figure 2**.

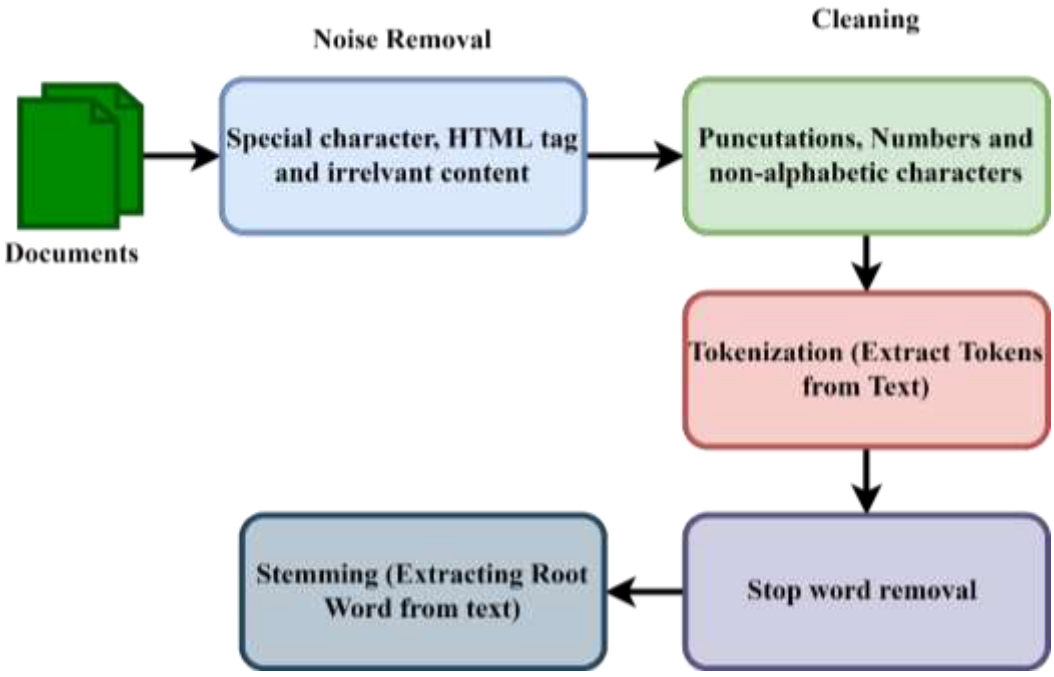


Figure 2. Process of text cleaning.

Initially, the non-textual information, unwanted data, HTML tags, and special characters are examined to eliminate the irrelevant details. The unwanted information was removed using regular expression analysis, HTML tag removal, and special character elimination. The numbers, non-alphabetic characters, punctuations, and undesirable information have to be removed from the text. Next, the Natural Language Toolkit (NLTK) is applied to the text for dividing the text into individual words or tokens. Further data dimensionality is reduced by removing the stopwords such as "the", "is", "in", and "and".

At last, stemming is done in which root words are extracted from the sentence by applying the Porter Stemmer Took kit. The stemming process reduces the computation complexity while performing the document clustering. The preprocessing process reduces the overfitting issues and maximizes the clustering accuracy. The noise-removed texts are fed into the feature extraction process to convert the text into a numerical representation. This study uses the Term Frequency and Inverse Document Frequency (TF-IDF) procedure to determine the critical terms in the documents. This process creates the document term metrics that have rows and columns; rows related to documents and columns have unique term information. The collected rows and column information is named TF-IDF scores. Then, the scores are normalized according to the document length. The documents-statistical information has to be extracted for understanding the document characterisitcs. Initially, the TF value is estimated according to the occurrences of the term (word), which is denoted as Equation (1).

$$TF(t,d) = \frac{Number\ of\ times\ terms\ t\ presented\ in\ d\ (document)}{Total\ number\ of\ terms\ in\ d} \quad (1)$$

In Equation (1), document term frequency $TF(t,d)$ is estimated by taking the ratio between the $\frac{Number\ of\ times\ terms\ t\ presented\ in\ d\ (document)}{Total\ number\ of\ terms\ in\ d}$. If the computed $TF(t,d)$ value is high, and then the particular word appears regularly in the document. After that inverse document frequency (IDF) is estimated using Equation (2).

$$IDF\ (t,d) = \log\left(\frac{Total\ numbe\ rof\ documnets\ in\ the\ corpus\ |D|}{Number\ of\ documents\ containing\ term\ t + 1}\right) \quad (2)$$

According to Equation (2), the inverse document frequency $IDF\ (t,d)$ value is estimated by taking the logarithmic ratio between the $\frac{Total\ numbe\ rof\ documnets\ in\ the\ corpus\ |D|}{Number\ of\ documents\ containing\ term\ t+1}$. From the computed TF and IDF values, the local and global features are extracted using Equation (3).

$$TF - IDF(t,d,D) = TF(t,d) * IDF(t,D) \tag{3}$$

If the computed $TF - IDF(t,d,D)$ value is high; the document consists of maximum terms and rates in the whole corpus. The extracted features are utilized for the clustering process to group similar documents. If the system receives a large volume of data, the number of features is also maximum, which leads to computation difficulties and uncertainty issues. The research difficulties are overcome by applying the feature selection approach, which is described in the below section.

Feature selection using bacterial foraging and rough set analysis (BF-RSA)

Feature selection is the process of selecting the optimal features from the feature list, which helps to convert the text data into processing format. During the analysis, the most relevant features or terms are retained, and irrelevant features are eliminated from the list. The effective incorporation of this step maximizes the clustering accuracy. This study uses the TF-IDF features for analyzing similar documents. The extracted features are explored using the Bacterial Foraging Optimization (BFO) approach. The BFO approach works according to the *E. coli* behavior that has four operations such as swarming, chemotaxis, elimination-dispersal and reproductions. The parameter utilized in the BFO process is listed in **Table 1**.

Table 1. Parameter utilized in BFO.

Parameter	Descriptions
N	Number of bacteria $i = 1,2,3, \ldots . N$
P	Search space dimension
N_{re}	Reproduction step number $K = 1,2, \ldots \ldots N_{re}$
N_{ed}	Count of Elimination-dispersal steps $l = 1,2, \ldots \ldots N_{ed}$
N_c	Count of chemotaxis step $j = 1,2, \ldots \ldots N_c$
N_s	Count of swimming steps $s = 1,2, \ldots \ldots N_s$
$\Delta(i)$	Random direction vector range $(-1,1)$
$C(i)$	Size of Chemotaxis step
P_{ed}	Probability of elimination-dispersal
$J(i,j,k,l)$	Fitness value of bacterium i^{th} chemotaxis j^{th} reproduction k^{th}, elimination dispersal l^{th}.

The first step of this work is chemotaxis, in which bacteria location and motion are stimulated. The bacteria direction is changed continuously to step size. Considering, that the bacteria identify the rich nutrients, then it has to be moving in the same direction. Assume θ is the bacteria position, which has i^{th} chemotaxis j^{th} reproduction k^{th}, elimination dispersal l^{th} which is denoted as $\theta^{i}(j,k,l)$. Then, the bacteria swimming process is defined in Equation (4).

$$\varphi(i) = \frac{\Delta(i)}{\sqrt{\Delta(i)^T \Delta(i)}} \tag{4}$$

In Equation (4), the random vector is represented as $\Delta(i); i = 1,2,3, \ldots, S$, bacteria count is denoted as S, every element in $\Delta_m(i)$ having the number between -1 to 1. For every swimming, the bacteria position has to be updated according to equation (5).

$$\theta^i(j+1,k,l) = \theta^i(j,k,l) + C(i)\varphi(i) \tag{5}$$

In Equation (5), the swimming phase moving step size is represented as $C(i); i = 1,2, \ldots\ldots S$..

After updating the bacteria position, bacteria moved from one location to another to get the nutrient food. During this process, bacteria proclamations attractant signal that helps to identify another cell in the search space. If bacteria identify high-nutrient food, they release chemical substances, else release repel signals to each other. Then, the swarming process is defined in Equation (6).

$$J_{cc}(\theta, P(j,k,l)) = \sum_{i=1}^{N} J'_{cc}(\theta, \theta'(j,k,l)) \tag{6}$$

Equation (6) has been further derived as follows.

$$\begin{aligned} J_{cc}(\theta, P(j,k,l)) = & \sum_{i=1}^{N}\left[-d_{attract}\, exp\left(-w_{attract}\sum_{m=1}^{P}\left(\theta_m - \theta_m^i\right)^2\right)\right] \\ & + \sum_{i=1}^{N}\left[h_{repellant}\exp\left(-w_{repellant}\sum_{m=1}^{P}\left(\theta_m - \theta_m^j\right)^2\right)\right] \end{aligned} \tag{7}$$

In Equation (12), the swarming process output is represented as $J_{cc}(\theta, P(j,k,l))$ which is added to the chemotaxis operation objective function value. Attractant signal parameters are $d_{attract}$ and $w_{attract}$; repellent signal parameters are represented as $h_{repellant}$ and $w_{repellant}$. After finishing the chemotaxis steps N_c reproduction steps are applied to the search space to predict the best solution. In this step, the bacteria's accumulated fitness value is estimated to identify the bacteria's health condition. If the bacteria has maximum fitness value, then the bacteria is unhealthy (minimum nutrition problem); therefore, that particular bacteria does not have a chance to reproduce. The computed fitness values are sorted in ascending order and divided into two halves. The first part is utilized for generation, and the next part is utilized to keep the population size. Finally, elimination-dispersal steps are taken into account when the bacteria meet a harsh environment like temperature change etc. In this stage, a few bacteria are selected according to probability value, and some bacteria are generated randomly in search space. The elimination-dispersal process entails the adjustment of the position of the deleted bacteria B_i as outlined below:

$$X_{ij}^{new} = X_{ij}^{old} + random().(X_{max} - X_{min}) \tag{8}$$

In Equation (8), minimum and maximum values in solution space are represented as X_{min} and X_{max}. The chemotactic parameters, including the step size and the number of chemotactic steps, should be modified under the bacteria's performance in locating improved solutions. According to the above procedure, the BFO Pseudocode is defined in Algorithm 1.

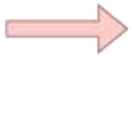
According to the algorithm steps, the optimized features are selected, which helps to perform the document clustering process. The above-initialized bacteria populations are denoted as the candidate features subset. During the analysis, a rough set approach is applied to develop the discernibility matrix, which helps to predict the upper and lower approximations. After that, the fitness function is identified to determine the clustering-related objective functions. The selected objective function helps to improve clustering quality, separations, and compactness. For every iteration, the bacteria position is updated according to the elimination-dispersal, reproduction, and chemotaxis. This iteration is performed until it reaches the stopping criteria and the best features are selected using the objective function.

Algorithm 1 Pseudocode of BFO algorithm

Initialize Parameters defined in **Table 1**.
Chemotaxis loop $j = 1,2, \ldots . . N_c$
Reproduction loop $k = 1,2, \ldots \ldots N_{re}$
Elimination-dispersal loop $l = 1,2, \ldots . N_{ed}$
$For\ i = 1,2, \ldots . N$
    Compute fitness function: $J(i,j,k,l) = J(i,j,k,l) + J_{cc}^{i}(\theta'(j,k,l), P(j,k,l))$
    Best fitness is denoted as: $J_{last} = J(i,j,k,l)$ which is computed from run.
    Produce random vector $\Delta(i)$
    Move one place to another place:
        $\theta^i(j+1,k,l) = \theta^i(j,k,l) + C(i)\frac{\Delta(i)}{\sqrt{\Delta(i)^T\Delta(i)}}$ to updated position $\theta^i(j+1,k,l)$
    Set S=0// counter of swim step
        $while\ s < N_s$
        $s = S = 1$
        $if\ J(i,j+1,k,l) < J_{last}$ then update $J_{last} = J(i,j+1,k,l)$
        Else $s = N_s$
        End while
Go to the next $i+1$ bacteria and perform for loop until to reach $i == N$
$If\ j < N_c$ Move to the chemotaxis loop.
The reproduction step is performed.
$If\ k < N_{re}$ Move to the reproduction loop.
Elimination dispersal is performed.
$If\ l < N_{ed}$ Move to elimination-dispersal loop; otherwise, end loop.

A rough set is one of the effective computational analysis approaches used to handle the vagueness and uncertainty issues while selecting a feature subset. The approach selects the features according to the boundary approximation in which lower and upper boundary approximations are selected. The selected approximation value enables to handle the imprecision and incomplete data. The data are characterized as the information system, which is defined as $\mathrm{U} = (\mathrm{X}, \mathrm{Y}, \mathrm{V}, \mathrm{f})$ in which X is defined as set of features , Y is represented as set of decision class, universe of discourse is denoted as V and mapping is denoted as f. During the mapping process, every object in $\mathrm{V}$is mapped with X attributes and Y decision class. The mapping equivalence relation is created with the help of indiscernibility (two objects share the same values for entire attributes). For the given feature subset $\mathrm{A} \subseteq \mathrm{X}$, the lower approximation value $[\mathrm{A}]_{\mathrm{X}}^{-}$ is set attributes in V. The upper approximation $[\mathrm{A}]_{\mathrm{X}}^{+}$ is the set of attributes in V. These attributes are widely applied to derive the feature subset from the extracted feature list. The discernibility matrix is one of the main tools to decide on feature selection. The matrix consists of attributes that are computed by comparing the dataset objects. In the matrix, the row is related to objects, and the column belongs to features; therefore, matrix entries are relevant to the object attributes. The attributes are determined by making the pair comparison and the two objects having different values. Then the matrix entry is denoted as (i, j), and the value is one if (i, j) objects have dissimilar values (discernible) and 0 otherwise. Let's assume the dataset has A, B and C objects with X and Y attributes. First, make the pairwise comparison to generate the discernible matrix. Compare A and B; here, X attribute values are the same, whereas *Y* and *Z* are dissimilar values. Then Compare *A* and *C* objects in which *X* and *Y* values are dissimilar and *Z* values also different. Finally, compare *B* and *C* objects in which *Y* and *Z* values are the same and *X* values are different. Then, the discernible matrix for a particular dataset is mentioned in **Figure 3**.

Objects	Attributes	
A	X	Y
B	X	Z
C	Y	Z



	A	B	C
A	0	1	1
B	1	0	0
C	1	0	0

Figure 3. Representation of discernible matrix.

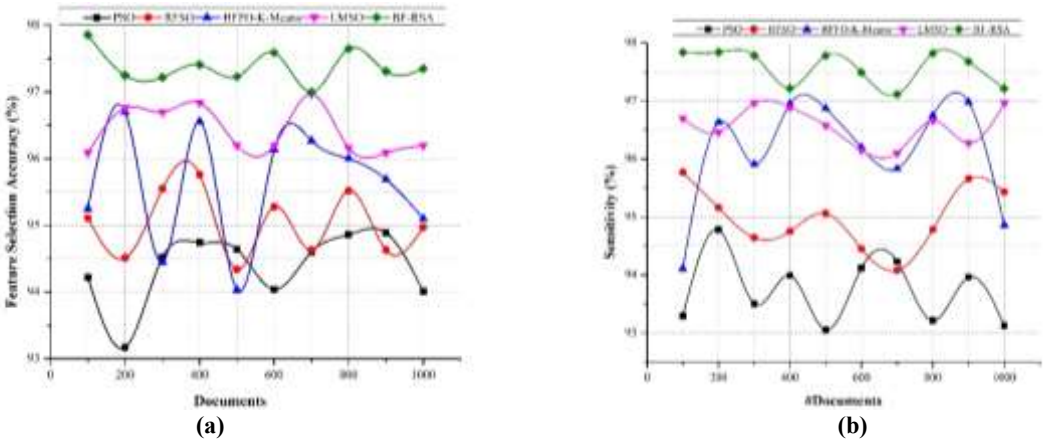
Figure 3 is represented as the discernible matrix in which one is denoted as the discernible of corresponding objects with at least one attribute. 0 represented as objects are not discernible. The generated discernible matrix is used to determine the redundant features and helps to preserve the subset features. From the derived minimal reduct features, a decision rule is generated that is relevant to the relationship between decision classes and selected features. The discernibility matrices are further explored using an upper and lower approximation to identify the optimized features. This process overcomes the vagueness and uncertainty issues in the document analysis. The selected features are fed into the K-means clustering algorithm to group similar documents in the same group. During the clustering process cluster centroid is estimated using Equation (9).

$$C_i' = \frac{\sum_{j=1}^{N} \mu_{ij} . D_j}{\sum_{j=1}^{N} \mu_{ij}} \tag{9}$$

In addition, the membership value is estimated $\mu_{ij} = \frac{1}{1+\frac{l_a \text{ of } C_i}{U_a \text{ of } C_i}}$ which helps to predict similar documents. Then, update the cluster centroid frequently to minimize the deviations while clustering the documents. Thus, the feature selection approach based on selected choosen attributes improves the overall clustering accuracy and minimizes the deviation between the outputs while handling high-dimensional data. Then, the efficiency of the system is evaluated using experimental results.

4. Results and discussions

The efficiency of the Bacterial Foraging and Rough Set Analysis (BF-RSA) feature selection-based document clustering is discussed in this section. As mentioned earlier, BBC dataset information is utilized for selecting the optimized featues. The gathered details are processed using the NLP technique that minimizes the overfitting issues by identifying the irrelevant information. Then stemming process is applied to derive the root word from the text which is fed into the TF-IDF feature extraction process that extracts the document's key features. A bacterial foraging optimization approach processes the extracted features. The adaptability and food-searching behavior identify the optimized feature list. Then discernibility matrix is generated to eliminate the redundant features. This process improves the overall feature selection accuracy and minimizes the difficulties in high-dimensional data analysis efficiency. Finally, lower and upper approximation criteria are fed into selecting the optimized feature set. The selected features are analyzed using k-means clustering that groups similar features with minimum deviation errors. The discussed system efficiency is compared with existing methods such as particle swarm optimization (PSO)[32], random feature set generation approach (RFSG)[20], Hybrid Fruit Fly Optimized K-means (HFFO-K-Means)[22], and Link-Based Multi-Sever Optimization Technique (LMSO)[30]. Then, the efficiency of selected features is evaluated, and the obtained results are illustrated in **Figure 4**.



(a) **(b)**

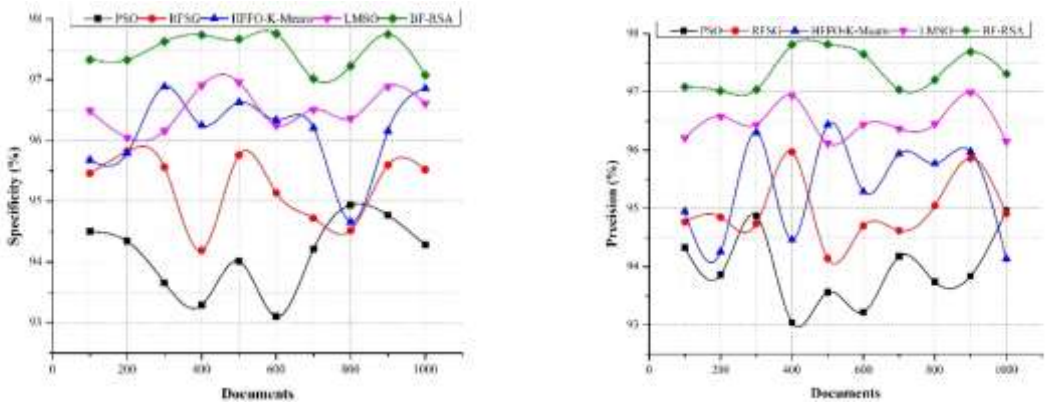


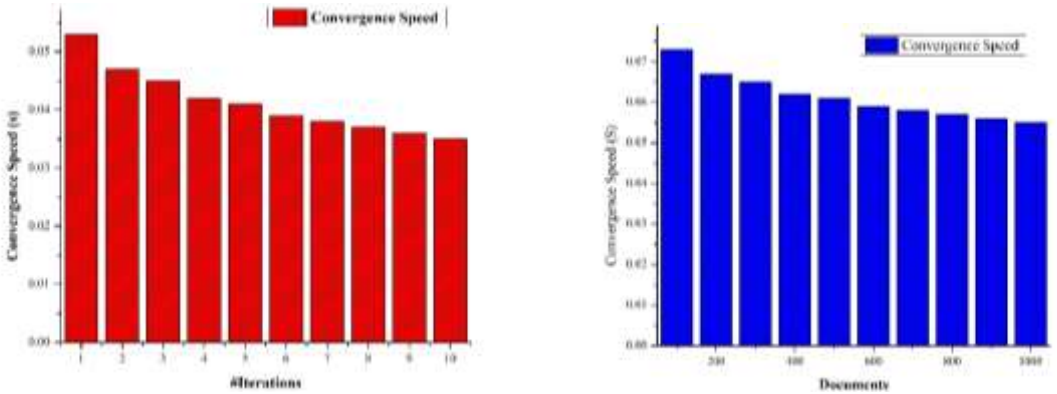
Figure 4. Feature selection efficiency analysis **(a)** accuracy; **(b)** sensitivity; **(c)** specificity; **(d)** precision.

Figure 4 illustrates the efficiency of Bacterial Foraging and Rough Set Analysis (BF-RSA) feature selection-based document clustering. The graphical analysis clearly shows that the BF-RSA approach attains maximum accuracy (97.54%) (**Figure 4a**) compared to existing methods. The high accuracy is directly related to maximum clustering accuracy and quality. The BFO algorithm identifies optimal solutions by exploring each feature in the search space. During the analysis, the algorithm uses the fitness function to identify the combination of features. The feature space is examined using a rough set approach that generates the discernibility matrix to identify the pairwise relationship between the features. According to the relationship, similar and dissimilar features are identified. The identified features minimize the uncertainty and approximation issues. The rough set approach is able to handle the noisy and irrelevant information that minimizes the redundancy issues, which leads to maximizing the sensitivity of the feature selection (97.46%) (**Figure 4b**). During the feature analysis, bacterial foraging optimization and rough set approach are integrated to improve the overall feature selection accuracy. The integrated approach identifies the optimal solution by fine-tuning the fitness function parameters. The fitness function helps to predict the appropriate feature with a minimum false negative and false positive value that directly indicates the system ensures maximum specificity (97.53%) (**Figure 4c**) and sensitivity values (97.46%). Finally, the rough set approach uses the lower and upper approximation values, which identify the optimal features and minimize the irrelevant features involvement in the document clustering. Therefore, Bacterial Foraging and Rough Set Analysis (BF-RSA) ensures high precision values (97.5%) (**Figure 4d**) compared to other methods. Further, the excellency of BF-RSA self-analysis is shown in **Table 2**.

Table 2. Feature selection efficiency of ERK-BFO.

Documents	Sensitivity	Specificity	Precision	Accuracy
100	98.02	97.46	98.2	97.89
200	98.22	97.24	98.59	98.02
300	98.93	97.32	98.71	98.32
400	98.29	97.95	98.36	98.2
500	98.81	97.3	98.12	98.08
600	98.08	97.69	97.02	97.6
700	98.95	97.59	98.33	98.29
800	98.93	97.82	98.21	98.32
900	98.76	97.4	98.76	98.31
1000	98.53	97.54	98.93	98.33

The self-analysis of the Bacterial Foraging and Rough Set Analysis (BF-RSA) based feature selection process ensures the high selection accuracy and maintaining consistent while examining the feature list (**Table 2**). The BF-RSA approach attains 98. 04% accuracy, which indicates that the method recognizes the features by maintaining reliability because the method analyzes entire features in the feature space. In addition, the rough set approach examines the boundary approximation that eliminates the irrelevant and inconsistent information that reduces the high-dimensionality issues. In addition, the BFO approach uses adaptability characteristics and food-searching behavior while exploring the feature set. The effective computation of pairwise comparison is used to identify the dissimilar features that help to maximize the overall clustering efficiency and quality. Similarly, the BF-RSA approach attains high specificity (98.59%), recall (97.60%), and precision (97.84%). The findings indicate that BF-RSA performance is reliable and of high quality, regardless of the size of the datasets. The approach is dependable for effective features because of its reliability and effectiveness in discovering significant document patterns. The effective integration of the BFO and RSA approach minimizes the convergence speed while exploring the huge volume of data. The convergence speed represented how effectively and fastly the possible features are selected while forming the feature subset. Then, the graphical analysis of BF-RSA-based convergence speed is illustrated in **Figure 5**.



(a) #Iterations. **(b)** #Documents.

Figure 5. Convergence speed analysis of BF-RSA.

Figure 5 illustrates the Convergence speed of the BF-RSA method while analyzing the large volume of data in the search space. The bacterial foraging optimization approach uses chemotaxis and swimming operations that identify the optimal solution according to the high-nutrient food (high fitness value). According to the high probability or fitness values the optimal features are selected. The selected features effectively contribute to clustering quality and accuracy. In addition, the rough set approach discernible and approximation criteria maximize the feature selection efficiency by addressing the convergence problems. The Bacterial Foraging Optimization algorithm balances exploration and exploitation, enabling it to adapt to the document dataset's specific properties dynamically. Then, the efficiency of the convergence speed is compared with the existing methods, and the obtained results are illustrated in **Table 3**.

Table 3 compares the convergence speeds of several feature selection algorithms. The table shows that the Bacterial Foraging Optimization Technique (BF-RSA) approach efficiency is evaluated for various numbers of documents. During the analysis, system efficiency is compared with different algorithms such as PSO, RFSG, HFFO-K-Means, and LMSO. From the comparison, the BF-RSA method addresses the high-dimensionality, convergence, and clustering quality-related problems. The fast convergence of BF-RSA may be credited to its technical elements, which involve handling uncertainty in traditional document clustering processes and achieving an optimal trade-off between exploration and exploitation in Bacterial Foraging Optimization. The findings of this study indicate that the BF-RSA algorithm can serve as an efficient and scalable approach for document clustering. According to the analysis, the ERK-BFO approach attains

97.50% accuracy when compared to the other methods, such as PSO (94.48%), RFSG (95.11%), HFFO-K-Means (95.70%), and LMSO (96.45%).

Table 3. Convergence speed analysis (s).

Documents	PSO	RFSG	HFFO-K-Means	LMSO	BF-RSA
100	0.39	0.32	0.28	0.26	0.073
200	0.37	0.3	0.25	0.23	0.067
300	0.34	0.27	0.24	0.21	0.065
400	0.33	0.24	0.22	0.2	0.062
500	0.31	0.23	0.21	0.18	0.061
600	0.3	0.22	0.2	0.16	0.059
700	0.25	0.2	0.18	0.15	0.058
800	0.24	0.18	0.16	0.14	0.057
900	0.23	0.16	0.14	0.13	0.056
1000	0.21	0.15	0.13	0.12	0.055

5. Conclusion

Thus, the study focuses on the Bacterial Foraging and Rough Set Analysis (BF-RSA) based feature selection for document clustering. Initially, the documents are collected from BBC dataset information, which is analyzed using the NLP process to eliminate the irrelevant information that is used to reduce the overfitting issues. Then TF-IDF process is applied to extract the terms from the document, which is a representation of the text. The derived features are explored using the rough set approach that generates the discernibility matrix which helps to identify the dissimilar features. After that, the bacterial foraging optimization approach is applied to predict the optimal feature. The BF algorithm uses the chemotaxis, swimming, elimination-dispersal, and reproduction steps to predict the optimal features. During the analysis, the fitness function is computed, which determines the most relevant features. The selected features are further explored using lower and upper approximation criteria that identify optimal features with maximum selection rate. (97.5%). However, the system requires training and learning to reduce the computation difficulties.

Author contributions

Conceptualization, SP; methodology, RK; software, SP, and RK; validation, RK; formal analysis, SP; investigation, SP; resources, SP; data curation, SP, and RK; writing—original draft preparation, SP; writing—review and editing, SP, and RK. All authors have read and agreed to the published version of the manuscript

Conflict of interest

The authors declare no conflict of interest.

References

1. Lydia EL, Moses GJ, Varadarajan V, et al. Clustering And Indexing Of Multiple Documents Using Feature Extraction Through Apache Hadoop On Big Data. Malaysian Journal of Computer Science. 2020; 108-123. doi: 10.22452/mjcs.sp2020no1.8
2. Abualigah L, Gandomi AH, Elaziz MA, et al. Nature-Inspired Optimization Algorithms for Text Document Clustering—A Comprehensive Analysis. Algorithms. 2020; 13(12): 345. doi: 10.3390/a13120345
3. Alguliyev RM, Aliguliyev RM, Isazade NR, et al. COSUM: Text summarization based on clustering and optimization. Expert Systems. 2018; 36(1). doi: 10.1111/exsy.12340

4. Jacksi K, Salih N. State of the art document clustering algorithms based on semantic similarity. Jurnal Informatika. 2020; 14(2): 58. doi: 10.26555/jifo.v14i2.a17513
5. Yang W, Wang X, Lu J, et al. Interactive Steering of Hierarchical Clustering. IEEE Transactions on Visualization and Computer Graphics. 2021; 27(10): 3953-3967. doi: 10.1109/tvcg.2020.2995100
6. Greene D, Cunningham P. Practical solutions to the problem of diagonal dominance in kernel document clustering. Proceedings of the 23rd international conference on Machine learning - ICML '06. Published online 2006. doi: 10.1145/1143844.1143892
7. Rose RL, Puranik TG, Mavris DN. Natural Language Processing Based Method for Clustering and Analysis of Aviation Safety Narratives. Aerospace. 2020; 7(10): 143. doi: 10.3390/aerospace7100143
8. Mora L, Deakin M, Reid A. Combining co-citation clustering and text-based analysis to reveal the main development paths of smart cities. Technological Forecasting and Social Change. 2019; 142: 56-69. doi: 10.1016/j.techfore.2018.07.019
9. Alowaimer BH, Dahiya D. Performance Investigation of Phishing Website Detection by Improved Deep Learning Techniques. Wireless Personal Communications. 2023; 132(4): 2625-2644. doi: 10.1007/s11277-023-10736-2
10. Chen J, Kudjo PK, Mensah S, et al. An automatic software vulnerability classification framework using term frequency-inverse gravity moment and feature selection. Journal of Systems and Software. 2020; 167: 110616. doi: 10.1016/j.jss.2020.110616
11. Borrelli D, Svartzman GG, Lipizzi C. Correction: Unsupervised acquisition of idiomatic units of symbolic natural language: An n-gram frequency-based approach for the chunking of news articles and tweets. PLOS ONE. 2021; 16(1): e0245404. doi: 10.1371/journal.pone.0245404
12. Milička J, Cvrček V, Lukešová L. Modelling crosslinguistic n-gram correspondence in typologically different languages. Languages in Contrast. 2021; 21(2): 217-249. doi: 10.1075/lic.19018.mil
13. Benabdellah AC, Benghabrit A, Bouhaddou I. A survey of clustering algorithms for an industrial context. Procedia Computer Science. 2019; 148: 291-302. doi: 10.1016/j.procs.2019.01.022
14. Fuchs M, Höpken W. Clustering: Hierarchical, k-Means, DBSCAN. In: Applied Data Science in Tourism: Interdisciplinary Approaches, Methodologies, and Applications. Springer International Publishing, Cham; 2020. pp. 129–149.
15. Ghosal A, Nandy A, Das AK, et al. A short review on different clustering techniques and their applications. Emerging Technology in Modelling and Graphics. In: Proceedings of IEM Graph. 2018. pp. 69–83.
16. Albalawi R, Yeap TH, Benyoucef M. Using Topic Modeling Methods for Short-Text Data: A Comparative Analysis. Frontiers in Artificial Intelligence. 2020; 3. doi: 10.3389/frai.2020.00042
17. Chen Y, Li CG, You C. Stochastic Sparse Subspace Clustering. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Published online June 2020. doi: 10.1109/cvpr42600.2020.00421
18. Baradaran AA, Navi K. HQCA-WSN: High-quality clustering algorithm and optimal cluster head selection using fuzzy logic in wireless sensor networks. Fuzzy Sets and Systems. 2020; 389: 114-144. doi: 10.1016/j.fss.2019.11.015
19. Abualigah LM, Khader AT, Hanandeh ES. A new feature selection method to improve the document clustering using particle swarm optimization algorithm. Journal of Computational Science. 2018; 25: 456-466. doi: 10.1016/j.jocs.2017.07.018
20. Christy A, Gandhi GM. Feature selection and clustering of documents using random feature set generation technique. In: Advances in Data Science and Management. Springer Singapore; 2020. pp. 67–79.
21. Lakshmi R, Baskar S. DIC-DOC-K-means: Dissimilarity-based Initial Centroid selection for DOCument clustering using K-means for improving the effectiveness of text document clustering. Journal of Information Science. 2018; 45(6): 818-832. doi: 10.1177/0165551518816302
22. Bezdan T, Stoean C, Naamany AA, et al. Hybrid Fruit-Fly Optimization Algorithm with K-Means for Text Document Clustering. Mathematics. 2021; 9(16): 1929. doi: 10.3390/math9161929
23. Kim H, Kim HK, Cho S. Improving spherical k-means for document clustering: Fast initialization, sparse centroid projection, and efficient cluster labeling. Expert Systems with Applications. 2020; 150: 113288. doi: 10.1016/j.eswa.2020.113288
24. Wang H, Zhou C, Li L. Design and Application of a Text Clustering Algorithm Based on Parallelized K-Means Clustering. Revue d'Intelligence Artificielle. 2019; 33(6): 453-460. doi: 10.18280/ria.330608
25. Abualigah L, Diabat A, Geem ZW. A Comprehensive Survey of the Harmony Search Algorithm in Clustering Applications. Applied Sciences. 2020; 10(11): 3827. doi: 10.3390/app10113827
26. Cekik R, Uysal AK. A novel filter feature selection method using rough set for short text data. Expert Systems with Applications. 2020; 160: 113691. doi: 10.1016/j.eswa.2020.113691
27. Abualigah L, Gandomi AH, Elaziz MA, et al. Advances in Meta-Heuristic Optimization Algorithms in Big Data Text Clustering. Electronics. 2021; 10(2): 101. doi: 10.3390/electronics10020101
28. Ibrahim RK, Zeebaree SRM, Jacksi KFS. Survey on Semantic Similarity Based on Document Clustering. Advances in Science, Technology and Engineering Systems Journal. 2019; 4(5): 115-122. doi: 10.25046/aj040515
29. Chen K, Zhou FY, Yuan XF. Hybrid particle swarm optimization with spiral-shaped mechanism for feature selection. Expert Systems with Applications. 2019; 128: 140-156. doi: 10.1016/j.eswa.2019.03.039

30. Abasi AK, Khader AT, Al-Betar MA, et al. Link-based multi-verse optimizer for text documents clustering. Applied Soft Computing. 2020; 87: 106002. doi: 10.1016/j.asoc.2019.106002
31. Hassani H, Beneki C, Unger S, et al. Text Mining in Big Data Analytics. Big Data and Cognitive Computing. 2020; 4(1): 1. doi: 10.3390/bdcc4010001
32. Abualigah LM, Khader AT, Hanandeh ES. A new feature selection method to improve the document clustering using particle swarm optimization algorithm. Journal of Computational Science. 2018; 25: 456-466. doi: 10.1016/j.jocs.2017.07.018