Evolutionary Intelligence (2020) 13:497–510
https://doi.org/10.1007/s12065-019-00335-1

RESEARCH PAPER

Incremental document clustering using fuzzy-based optimization strategy

Madhulika Yarlagadda[1,2] · Gangadhara Rao Kancherla[3] · Srikrishna Atluri[2]

Received: 29 May 2019 / Revised: 15 November 2019 / Accepted: 11 December 2019 / Published online: 17 December 2019

Abstract
The technical advances in the information systems contribute towards the massive availability of the documents stored in the electronic database, such as e-mails, internet and web pages. Thus, it becomes a complex task for arranging and browsing the required document. This paper proposes an incremental document clustering method for performing effective document clustering. The proposed model undergoes three steps for document clustering, namely pre-processing, feature extraction and Incremental document categorization. The pre-processing step is carried out for removing the artifacts and redundant data from the documents by undergoing stop word removal process and stemming process. Then, the next step is the feature extraction based on Term Frequency-Inverse Document Frequency (TF–IDF) and Wordnet features. Here, the feature is selected using support measure named ModSupport, and then, the incremental document clustering is performed based on the hybrid fuzzy bounding degree and Rider-Moth Flame optimization algorithm (RMFO) using the boundary degree. Here, the RMFO aims at the selection of the optimal weights for the boundary degree model and is designed by integrating Rider Optimization Algorithm (ROA) with Moth Flame optimization (MFO). The performance of the proposed RMFO outperformed the existing techniques using accuracy, F-measure, precision, and recall with maximal values 93.98%, 94.876%, 93.958% and 93.964% respectively.

Keywords Incremental document · Stop word removal · Boundary degree · Stemming · Clustering

1 Introduction

The data is accumulated in different sets, which consist of documents. Each document comprises of the key-value pairs, and these values are compiled in nested subdocuments. The documents facilitate more flexibility in designing the schema, and they pose the capability to store multifaceted ordered data and assorted data into a single group. Moreover, the databases are affirmed as schema-less, and the schemas are used for conveying the data model [1]. The progression in technologies and organizations collects huge amounts of data, which poses different structures, speed, and types. The alteration of the proper way may lead to huge growth and enables the organizations to entirely change the information in the growing world that necessitates advanced data analysis and help to enable the innovative systems for dealing with huge data and lack of structures [2]. The database offers a wrapper that can be utilized for retrieving the information with respect to the given query. The documents that are to be retrieved can be accessed from these huge datasets using the queries. In conventional databases, the documents are handled in a particular manner and content of the particular field is directly accessed on the basis of queries. The queries are formulated, and results are retrieved based on indexing relations amongst different fields [3].

The growing size of the documents persuades rehabilitated interest in retrieving huge documents. The existing methods enhance the retrieval results, and the documents

✉ Madhulika Yarlagadda
madhulika.yarlagadda@gmail.com

Gangadhara Rao Kancherla
kancherla123@gmail.com

Srikrishna Atluri
atlurisrikrishna@gmail.com

[1] Department of Computer Science and Engineering, JNTUK, Kakinada, Andhra Pradesh, India

[2] Department of Information Technology, RVR&JC College of Engineering, Chowdavaram, Guntur, Andhra Pradesh, India

[3] Department of Computer Science and Engineering, Acharya Nagarjuna University, Guntur, Andhra Pradesh, India

obtained through the retrieval results are precise and plays a significant role while retrieving documents from long and structured documents. Huge documents with multifarious structures and short documents briefing multiple subjects are the major challenges while retrieving the documents. The method that does not pose the ability to differentiate a few matches from a document may initiate several complexities while retrieving the documents and becomes complicated while retrieving huge documents [4]. Due to rapid increase in documents, the understanding of queries is important for obtaining the required information that is essential to fulfilling the needs of the users. The conventional systems for retrieving the information relies on matching keywords and is adapted for indexing the documents from the database in which the documents are symbolized using certain mechanisms, which involves Boolean model, vector space model and probabilistic model [5, 6], but the document clustering is an important problem while retrieving the information from huge documents. Moreover, the clustering of documents is considered as a significant tool for Web search engines to browse the required documents. The document clustering is utilized for learning information retrieval systems [7].

Document clustering is examined in different areas for retrieving crucial information. Previously, the document clustering was adapted to improve the precision or recall in different information retrieval techniques and is considered as an effective way of determining the relevant documents from the set of huge documents. Numerous clustering methods are devised for browsing the documents or arranging the results obtained by the search engine with respect to certain queries. K-means techniques and agglomerative hierarchical clustering are the commonly used methods for clustering the documents. Agglomerative hierarchical clustering is known for its quality, whereas a K-means technique is adapted due to its higher efficiencies [8]. However, the conventional methods are expensive and need more effort for clustering the documents. Incremental clustering methods offer a reliable solution for reducing the cost of computation and make it applicable to huge datasets [9]. The incremental clustering techniques are processed in two stages. The first stage updates the micro-clusters in a real-time manner, and then in second stage, the clustering is carried out using stored synopsis when the request is initiated by the user [10]. The incremental PLSA algorithm is devised for estimating the parameters that help to update the incremental hyperparameters. Moreover, the parameters update and integrate the previous hyperparameters with statistics that are based on the current documents [9].

This paper aims to devise an incremental document clustering method using the hybrid fuzzy bounding degree. The proposed model undergoes three steps for the document clustering, namely Pre-processing, feature extraction, and Incremental document categorization. In pre-processing the artifacts and redundant words from the document have been removed. Then, the next step is feature extraction using the TF–IDF and Wordnet features. The features are selected based on the support equation named ModSupport, and then, clustering is done using the Rn-MSA, which is the integration of ROA and the Moth Search Algorithm (MSA). Whenever new documents arrive then, the clustering of those documents is performed using the proposed incremental document clustering mechanisms. Here, the incremental document clustering is performed based on the hybrid fuzzy bounding degree that includes the Rider-Moth Flame Optimization algorithm (RMFO) and boundary degree. Here, the RMFO aims at the selection of the optimal weights for the boundary degree model. The RMFO is the combination of ROA and MFO.

Proposed RMFO algorithm using a fuzzy boundary degree: The proposed RMFO is the integration of ROA in MFO in such a way that the developed algorithm aims at formulating the optimal weights for the boundary degree model.

The paper is structured in the following manner: Sect. 1 provides the introductory part of the document clustering, and Sect. 2 discusses existing methods of Document clustering with challenges of the methods that remain the motivation for the research. The proposed method of routing is demonstrated in Sect. 3, and Sect. 4 demonstrates the results of the methods. At last, Sect. 5 concludes the research work.

2 Motivations

This section illustrates the review of the existing methods along with the demerits, which highlight the challenges of the work, serving as the motivation for proposing a novel incremental document clustering scheme.

2.1 Literature review

The review of the existing methods is given as follows: Sangaiah et al. [11] developed three methods, named semi-supervised techniques, semi-supervised with dimensionality reduction, and unsupervised techniques for constructing a clustering-based classifier using Arabic text documents. At first, the pre-processing was carried out for eliminating the stop words using incremental k-means and dimensionality reduction techniques. Then, a weighting method was applied for obtaining the weight of each term with respect to its corresponding document, and finally, a similarity measure method was utilized for measuring the similarities with respect to other documents. The method used entropy, F-measure, and support vector machine (SVM) for computing the accuracy. The datasets considered for the evaluation were online dynamic datasets, which were considered on the

basis of availability and credibility of internet. Moreover, the Arabic language was challenging while applied in inference-based algorithm. Thus, choosing a suitable dataset was a prime factor for considering this research. The accuracy of these methods was compared with other existing methods, and this method yielded improved accuracy with fewer errors for classification. However, the method was unable to determine the reduction ratio and failed to calculate Chi Square similarity. Kotte et al. [12] developed an advanced similarity function for clustering the patterns of features based on the classification. The similarity function was utilized for performing supervised learning based on dimensionality reduction. The features considered in this work was word distribution and dimensionality reduction. The method attained optimal dimensionality reduction and retained the word distribution with improved classification accuracies as compared to other measures and obtained better classification accuracies. However, the method failed to devise a new membership functions and was unable to apply them on clustering to acquire clusters with improved cluster quality. Wan et al. [10] developed a method named incremental clustering approach using Gaussian mixture model (GMM) named incremental construction of GMM tree (ICGT) for building the data clustering on the obtained data. The ICGT constructs and adjusts the GMM tree in a dynamic manner to sequentially present the data. Here, each leaf node of the tree matches the dense Gaussian distribution and each non-leaf node corresponds to the GMM. Moreover, the tree update algorithm was adapted for defining the connectivity among nodes. The method was effective in creating the clusters but failed to design a clustering evaluation criterion and a search scheme for determining the final data partition using the constructed GMM tree. Li et al. [9] developed a weighted incremental PLSA algorithm named WIPLSA for discovering the topics in a dynamic order and then incrementally learn the topics using the new documents. The experiments were conducted to validate that the WIPLSA was able to confine that the topics hidden in the dynamic updating data for mining huge data. The method proved to be better in bafflement of huge dataset, which made it relevant for big data mining. Moreover, the method provided improved performance in application to document categorization, but the method was unable to detail the data with particular applications and generalizations.

Mulay and Shinde [13] developed a method named correlation-Based incremental clustering algorithm (CBICA) for creating the clusters based on the data of diabetic patients and examining the relationships, which indicated the reason for the increase in diabetic level. The obtained results were compared with the existing methods in terms of accuracy parameter. The algorithm was parameter-free and the end user provides the input dataset, the clustering is performed automatically without any additional dependencies from the user and uses the assumption of centroids, cluster count, and distance measures for the evaluation. This method uses the new outliers to rank the clusters and its principal components. This approach provided poor scalability and accuracies and was unable to deal with other domains link juvenile diabetes or gestational diabetes. Madhusudhanan and Jaganathan [14] designed a framework named classification of unstructured data using incremental learning (CUIL), which clustered the metadata and assigned the label for each cluster and then constructed a model using extreme learning machine (ELM), and feed-forward neural network for each batch of the data. The method trained the batches in a separate manner and minimized the memory resources for providing greater accuracy and efficiency. However, the method contained certain limitations, which involved fixing the random weights that were difficult for huge training dataset. Moreover, the method failed to redesign the model to address the issues of concept drift for unstructured documents. Kannan et al. [15] developed a method for determining the events in real-time and from the Cricket sports domain. The feature vectors of the live tweets were constructed using the TF–IDF representation, and the tweeted clusters were determined using the locality sensitive hashing (LSH) in which the post rate of each cluster was devised on the basis of volume of tweets computed. If the rate of post was above the specified threshold then, the key event was recognized from that cluster with specific event lexicon. Moreover, the specified threshold was used to filter the small spikes. The method provided effective clusters but failed to speed up the process using other data structures. Liu et al. [16] developed two incremental fuzzy clustering algorithms based on dynamic time warping (DTW) distance. For employing single-pass and online patterns, these algorithms would handle huge-scale time-series by dividing it into different chunks, to process sequentially. Besides, these algorithms selected the DTW for measuring the distance of pair-wise time-series to provide higher clustering accuracy and in addition, to provide the optimal match between two time-series data. The method was better than other three algorithms based on F-Measure and Entropy and provided improved performance on these datasets.

2.2 Challenges

The challenges faced by the existing method are illustrated as follows:

- In [12], a similarity function was developed for clustering huge data. Here, the clustering is an unsupervised learning method, which is used to place similar entities at a single place. However, the clustering process is considered as a major challenge to attain precise results. Furthermore, the accuracy is a barrier, as there is no prin-

ciple to decide the correct clusters and efficiency is an obstacle due to the quality of the cluster, which affects the clustering performance.

- The two-stage strategy was employed in conventional methods for performing incremental clustering. In two-stage strategy, the first stage updates data in real-time, and in second stage, the clustering process is carried out for processing data in quicker manner. The method is effective in dealing with huge datasets but takes more time to provide the clustering results [10].
- The incremental learning is employed for training the models based on acquired data and knowledge. However, the categorization of the data is considered as a major challenge due to high variations and missed labels. The variations in the data limit the computational resources, like time and memory [14].
- In [1], DTW-based fuzzy clustering was developed using the time-series data and devised three substitutes. This method used stretching or compressed variations using temporal data and is enviable for fuzzy clustering of time-series, but this method is not applicable for huge-scale data processing.
- The TF–IDF was devised to determine key events in real-time. However, the method poses many challenges, which involve limited length and noises, like types, grammar errors, and becomes complicated while dealing with huge volume of data [15].

3 Proposed hybrid fuzzy boundary model for incremental document clustering

In this section, the proposed hybrid fuzzy boundary model is illustrated for performing incremental document clustering over large documents using a fuzzy bounding degree. In this model, the incremental clustering method is incorporated to provide higher cluster consistency, which is a brief statistical representation of pair-wise document similarities amongst the clusters. The clusters are required to uphold high cohesiveness, while new documents are being added. The model undergoes three stages for performing the document clustering, which involves pre-processing, feature extraction, and incremental document clustering. The pre-processing is carried out for yielding effective management of the proposed incremental document categorization, which helps to enable the document clustering of the dynamic data. In pre-processing, the stop word removal and stemming are carried out to speed up the task of clustering. At first, the keywords from the documents are subjected to pre-processing to eliminate redundant and superfluous words from the data using stop word removal and stemming. Thus, lot of CPU cycles and memory can be saved by pre-processing and gives better results. The next step is feature extraction using TF–IDF and WordNet for determining the keywords of the document and is highly recommended for the feature selection process. Then the clustering is performed using the Rn-MSA, and the final step is incremental document clustering, which is devised on the basis of a hybrid fuzzy bounding degree designed by combining RMFO and boundary degrees. The goal of the proposed RMFO is to select the optimal weights for processing boundary degree model, which is designed by integrating ROA [17] and MFO [18]. Here, the cluster-based indexing is carried out using the proposed RMFO, and hamming distance is employed for determining the similar documents on the basis of minimum distance measure, and finally, the document clustering is done. Figure 1 depicts the block diagram of the proposed method of incremental document clustering.

Consider the input data denoted as D containing various attributes, which is given as,

$$D = \{D_{gh}\};(1 \le g \le K)(1 \le h \le L) \tag{1}$$

where D_{gh} denote the documents present in the database D specifying hth attribute in gth data. Here, K data points are considered with L number of attributes for each data point. The next step is to pre-process the documents for easier processing.

3.1 Pre-processing

In this section, the pre-processing is considered as a significant step for organizing thousands of documents in a smooth manner to provide effective results. The pre-processing helps to describe the processing of documents to obtain better representations. The database consists of unnecessary words or phrases, which impacts the clustering process. Thus, pre-processing becomes an important process for eliminating inconsistent words from the database. In pre-processing, the stop words removal and stemming are carried for refining the documents.

(i) *Stop word removal* The stop words are the words, which do not carry any information. The stop word removal is the process of removing the stop words from the huge text documents. Here, the non-information behavior words are eliminated to reduce the noise contained in the data. The removal of stop words can be used for saving large-space and to making the processing faster for acquiring effective results. Here, the stop words, such as auxiliary verbs or meaningless words are eliminated from the document.

(ii) *Stemming* The stemming process is used to transform the words to its stem. In large documents, various words are utilized that conveys the same concept.

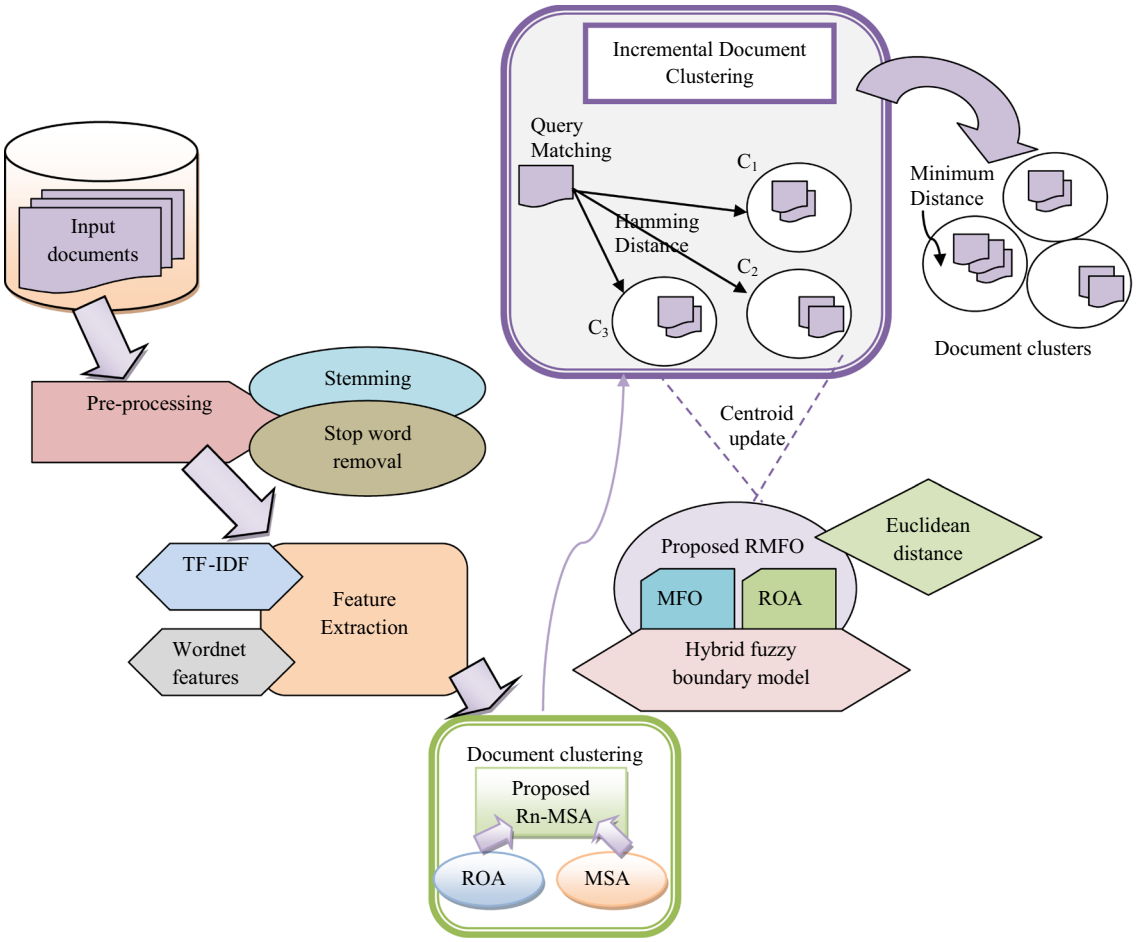

Fig. 1 Block diagram of proposed hybrid fuzzy boundary model using proposed RMFO for incremental document clustering

The significant technique used for reducing the words to its roots is called stemming. For instance agree, agreeing, and disagree belong to the word agree. The process of stemming is compact, easier to use, and is relatively accurate and does not require the suffix list.

3.2 Extraction of features for generating highly relevant features

The section deliberates the significant features extracted from the input document, and the significance of feature extraction is to generate the highly relevant features that enable the better clustering of the available documents. On the other hand, the complexity of analyzing the document is minimized as the document is represented as a reduced set of features. The feature extraction is carried out after pre-processing by removing the keywords from the documents using TF–IDF. TF is adapted for computing the frequency of occurrence of each word in a document. IDF is used for computing imperative word that rarely occurs in the document. Moreover, the accuracy associated with the clustering is assured through the effective feature extraction and the selected features using the input documents. Moreover, the feature extraction is initiated by parsing the input document using TF–IDF and WordNet strategies.

(a) *TF–IDF* The TF–IDF [11] is a measure, which aims to reveal the importance of the word in a document. The value of the TF–IDF increases with the number of words appears in the document and helps to adjust the appearance of words. The TF–IDF is formulated as,

$$T_h D_{a,g} = \frac{F_{g,a}}{\sum_{g=1}^{n} F_{g,a}} \times \log \frac{|D|}{|(D_g T_a \in D_g \in D)|} \tag{2}$$

where $F_{g,a}$ denote the frequency of term T_a in the document D_g and $\sum_{g=1}^{n} F_{g,a}$ is the total frequencies of all terms contained in the document D_g, and the term $D_g T_a \in D_g \in D$ represents the number of documents containing the term T_a.

(b) *WordNet* The WordNet [19] is utilized for measuring the relation between the terms from the given set of

words. The WordNet strategies can be utilized for computing the scores of semantic relations. The WordNet is used for covering a particular term from each subject related to its terms. WordNet is a lexical database that contains the standard documents into its specific lexical categories. WordNet is adapted for adding synonyms and hyponyms to improve the feature extraction process.

3.3 Feature selection based on ModSupport function for clustering

The next step is feature selection, which is an important step for solving the issues of document categorization. However, many feature selection methodologies are applied for feature selection. In this method, the selection of features is done using the ModSupport function. The function is utilized for evaluating the frequently appeared data items. It is a generic function to count the support of given data from a given database. Here, the support function is modified and is represented as,

$$S(W_j) = \frac{1}{t}\sum_{r=1}^{t}\frac{Y_r}{X_r} \times M_r \tag{3}$$

where t is the length of the frequent words, Y_r represents the total rules that are covered by W_r with rth length sequence, X_r is the total rules in the rth length sequences, and M_r is the average value of support of W_j in the rth sequences. Based on the ModSupport function, the top y features using the highest selection metric scores are selected to represent the intended document in the document representation phases. The selected features are represented at feature vector, which is provided to the clustering, in which the documents are grouped into various clusters.

3.4 Clustering using the Rn-MSA for the selection of the relevant documents

This section briefly explains the Rn-MSA [20] algorithm for clustering the documents. Rn-MSA is developed through the hybridization of the MSA and ROA, which aim at rendering the optimal centroids. Here, the update equation of ROA is integrated with the update equation of MSA for initiating the clustering process. Thus, the obtained equation of Rn-MSA algorithm is given by,

$$Y_{j,k}^{i+1} = \frac{1+\cos\left(P_{j,k}^{i}\right) * \partial_{j,k}^{i}}{1+\cos\left(P_{j,k}^{i}\right) * \partial_{j,k}^{i} - \lambda\beta}\left[(\lambda - \lambda\beta)Y_{j,k}^{i}\right] \tag{4}$$

where the steering angle of jth rider in kth coordinate is represented as $P_{j,k}^{i}$, and the distance travelled by jth rider in kth coordinate is denoted as, $\partial_{j,k}^{i}$. The scaling factor is denoted as λ, and the acceleration factor is denoted as β. The distance is measured based on the product of the off-time and velocity, and $Y_{j,k}^{i}$ denote the position of riders at current iteration. Based on the proposed Rn-MSA, the optimal centroids are derived. The clusters or the centroids determined using Rn-MSA is given as,

$$C = \left\{c_1, c_2, \ldots, c_i, \ldots, c_o\right\} \tag{5}$$

where c_i denote the ith cluster centroid, and o is the total centroids.

3.5 Incremental document clustering using proposed RMFO-based hybrid fuzzy boundary approach

The final step is incremental document clustering, which is devised based on hybrid fuzzy bounding degree that includes the RMFO and boundary degree, and particularly, RMFO aims to select the optimal parameters for the boundary degree model. The proposed RMFO is the combination of ROA [17] and MFO [18]. ROA is motivated by the behaviors of riders, who race toward the target position by updating its positions. The ROA is highly efficient and performs effectively in the fictional computing for solving the optimization issues, but possesses lower convergence and it is highly sensitive to the hyperparameters. The demerits of ROA are overcome using MFO that offers better convergence rate while obtaining global optimal solution. MFO is duly based on the behavior of the moth flames that changes its position for navigation. Moreover, it poses the ability to balance exploration and exploitation in a proper manner. Integrating the ROA with MFO outperforms the drawback associated with the standard optimizations, ROA and MFO regarding the optimal solution. Furthermore, the fuzzy bounding model is used for setting the boundary threshold in order to place the document in a specific cluster. The steps involved in the training algorithm are discussed below:

3.5.1 Index matching for document retrieval using hamming distance

When a keyword or document is given as the query, it is matched with the centroids initially and then, with each document in the corresponding cluster using Hamming distance [21]. Here, the centroids are formed using the Rn-MSA algorithm. At first, the features selected from the query document or query word are subjected to the matching process by calculating the Hamming distance between the query and the centroid. In the estimation, the Hamming distance is computed between the query and the individual centroids, and the centroid that acquired the minimal Hamming distance is selected as the best centroid in relevance with

the query. Then, the individual document in the selected centroid is matched with the query based on the hamming distance in such a way to retrieve the relevant documents. Thus, the incremental data is handled effectively based on the hamming distance using the incremental clustering approach named, RMFO-based fuzzy bounding approach. Accordingly, in the first-level match, the selection of the centroid is based on the minimal distance calculated between the centroid and the query. The second level of matching is done by computing the distance between all the documents in the matched centroid and the query. Thus, the documents relevant to the search or the input query are retrieved using the Hamming distance measure, given as,

$$D(A,B) = \sum_{i=1}^{n} M_A(c_i) - M_B(c_i) \tag{6}$$

where c_i is the current centroid, M_A represents the membership of A, and M_B denote the membership of B, and A and B indicates the centroid and individual document and it is peculiar to note that the membership functions are computed based on fuzzy. Here, the boundary is set to threshold 'T'. If the hamming distance, $D(A,B) < T$, or in other words, if the distance between two membership-based on fuzzy is less than the specified threshold, then the new document belongs to the cluster that rendered minimal distance, or else the new document is placed in new category cluster. The new category remains as another centroid such that the clustering in future includes the clustering based on the newly updated centroid in addition with the already available centroids, which marks the effective management of the incremental data.

3.5.2 Centroid update based on the proposed RMFO algorithm

The centroid update is essential for processing the incremental documents. As the important documents arrive incessantly without a particular boundary, managing the documents becomes complex. Thus, the incremental document clustering is employed for updating the cluster centroid based on proposed RMFO algorithm. For inheriting the centroid update, the obtained equation is given by,

$$C_i^{update} = \alpha C_i + \beta d^{new} \tag{7}$$

where C_i is the current centroid, and d^{new} is the new document. The value of α and β is the constants, which is determined optimally using the RMFO algorithm, which sets the effectiveness of the proposed RMFO-based fuzzy bounding approach.

Evaluation of fitness function The fitness function is computed for all the solutions to acquires the optimal values of α and β. The fitness is calculated between the centroid and the data point using the euclidean distance. The fitness constraint equation is given below,

$$Fitness = \sqrt{|(z_2 - z_1)|^2 + |(y_2 - y_1)|^2} \tag{8}$$

3.5.3 Proposed RMFO algorithm for centroid update

The incremental document clustering is performed by updating the centroids of the clusters using the proposed RMFO algorithm, which is designed by combining MFO and ROA. The proposed RMFO algorithm inherits the advantages of both ROA and MFO and provides best performance of document clustering. MFO is an algorithm developed based on the behavior of moths, while ROA is inspired on the basis of group of riders, which race towards a target location. Moreover, the ROA uses fictional computing and follows the procedure of fictional computing for solving the problems of optimization on the basis of imaginary ideas and thoughts. The inclusion of MFO in the update process of ROA enhances the convergence and thereby, improves the performance of the algorithm.

(a) *Computation of update equation*

Here, the update position of the overtaker is used in the update process for maximizing the success rate by determining the position of overtaker. ROA enhances the convergence and improves the performance of the algorithm. Thus, the expression according to the update process of the overtaker using the ROA is given by,

$$B_{v,w}(k+1) = B_{v,w}(k) + \left[F_v(k) * B_w^*\right] \tag{9}$$

where $B_{v,w}(k)$ represents the current solution, and $F_v(k)$ represents the direction pointer of vth rider at time k and B_w^* is the best solution.

The MFO algorithm updates the solution space on the basis of flame intensity. The alterations in the flame intensity make the moth move in one direction. Thus, the update solution of MFO is given by,

$$B_{v,w}(k+1) = E_v.e^{oq}.\cos(2\pi q) + P_w \tag{10}$$

where E_v is the distance between vth moth and wth flame, o represents the constant for describing the shape of the logarithmic spiral, q specifies the random number in the range $[-1, 1]$, P_w represents the wth flame. Equation (10) is modified with the ROA in order to enhance the effectiveness of the algorithm and to find solutions to various optimization issues, and the modified equation is given by,

$$B_w^* = \frac{B_{v,w}(k+1) - B_{v,w}(k)}{F_v(k)} \tag{11}$$

As P_w and B_w^* are same, both represent the best solution. Thus the Eq. (10) can also be written as,

$$B_{v,w}(k+1) = E_v.e^{oq}.\cos(2\pi q) + B_w^* \tag{12}$$

Substituting Eq. (11) in Eq. (12), the update equation derived is,

$$B_{v,w}(k+1) = E_v.e^{oq}.\cos(2\pi q) + \frac{B_{v,w}(k+1) - B_{v,w}(k)}{F_v(k)} \tag{13}$$

$$B_{v,w}(k+1) - \frac{B_{v,w}(k+1)}{F_v(k)} = E_v.e^{oq}.\cos(2\pi q) - \frac{B_{v,w}(k)}{F_v(k)} \tag{14}$$

After rearranging the above equation,

$$B_{v,w}(k+1)\left[1 - \frac{1}{F_v(k)}\right] = E_v.e^{oq}.\cos(2\pi q) - \frac{B_{v,w}(k)}{F_v(k)} \tag{15}$$

$$B_{v,w}(k+1)\left[\frac{F_v(k)-1}{F_v(k)}\right] = E_v.e^{oq}.\cos(2\pi q) - \frac{B_{v,w}(k)}{F_v(k)} \tag{16}$$

The final equation is given by,

$$B_{v,w}(k+1) = \frac{F_v(k)}{F_v(k)-1}E_v.e^{oq}.\cos(2\pi q) - \frac{B_{v,w}(k)}{F_v(k)-1} \tag{17}$$

The steps involved in the proposed RMFO are illustrated as follows:

Step 1: Initialization

The first step is to initialize the solution space with the position of the riders. The solution of ROA is in the form of a vector is given by,

$$B_k = \left\{B_{v,x}^k\right\};\ 1 \le v \le C;\ 1 \le x \le J \tag{18}$$

where B_{vx}^k represents the steering angle of vth rider vehicle at xth coordinate and C represents the total riders and the J denote total coordinates.

Step 2: Evaluation of success rate

The success rate or fitness of the solution is computed on the basis of the distance measure. The Euclidean distance is employed for computing the distance between the centroid and the document. Thus, the fitness of the solution is depicted in Eq. (8).

Step 3: Position update of the rider groups:

The bypass riders follow a common path without tracking the leading rider. The equation of the bypass rider is given by riders,

$$B_{v,x}^b(k+1) = \alpha\left[B_{\mu x}(k) * \gamma_x * \beta_{\vartheta,x}(k) * \left[1 - \gamma_x\right]\right] \tag{19}$$

where α denote random number ranging from 0 and 1, μ is random number, γ is a random number ranging from 0 and 1, and ϑ is a random number.

The attacker tends to seize the position of leaders by following the update process of leader, but the attackers update the values in the coordinates rather than the selected values, and thus, the update process of the attacker is given by,

$$B_v^a(k) = B_w^* + \left|\cos\left(W_{vw}(k)\right) * B_w^* * I_v(k)\right| \tag{20}$$

where B_w^* denote the position of the leading rider, $W_{vw}(k)$ denote the steering angle of the vth and wth coordinate, and $I_v(k)$ is the distance travelled by vth rider.

The follower tends to update the position using the position of leading rider in order to reach the target in a quicker manner and the equation of the follower is given by,

$$B_{v,w}^f(k+1) = B_w^* + \left|\cos(W_{vw}(k)) * B_w^* * I_v(k)\right| \tag{21}$$

where w is the coordinate selector, B_w^* is the best position, $W_{vw}(k)$ denote the steering angle of the vth and wth coordinate, and $I_v(k)$ is the distance travelled by vth rider.

Step 4: Determining the best solution

The solution is best if it acquired minimal fitness value. In addition, the update of the rider parameters is essential to determine the best solution.

Step 5: Termination

The steps are repeated until the iteration reaches the maximum count. Thus, the optimization renders the optimal values for the centroid update that is eminent to note that the proposed approach is effective in dealing with the incremental data.

4 Discussion of results

This section illustrates the results and discussion of the techniques for incremental document clustering is illustrated in this section with an effective performance analysis by comparing the methods with conventional methods.

4.1 Experimental Setup

The execution of the proposed technique is performed in the PC with Windows 10 Operating System, 2 GB RAM, Intel i-3 core processor. The proposed method is executed in the MATLAB platform.

4.2 Database description

The experimentation is conducted on two datasets, namely Reuter database and 20 Newsgroups database for performing the big data clustering.

4.2.1 20 Newsgroups database

This dataset [22] is taken from 20,000 newsgroup documents, which is divided into twenty different newsgroups. The dataset is collected by Ken Lang and became popular in the experimentation for text applications using machine learning techniques, like classification of texts and clustering texts.

4.2.2 Reuter database

The Reuter dataset [23] contains documents, which are linked with the stories of news. Here, the documents are partitioned into PEOPLES, TOPICS, ORGS, PLACES, and EXCHANGES.

4.3 Performance metric

The metrics employed for the analysis are F-measure, accuracy, recall, and precision they are formulated as,

Accuracy The accuracy indicates the accurate detection process that is calculated as,

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{22}$$

where TP is true positive, TN denote true negative, FP represents false positive and FN indicates false negative respectively.

F-measure It is defined based on the harmonic mean of the recall and precision for evaluating the clustering performance and is expressed as follows,

$$F_m = 2\frac{P_i * R_i}{P_i + R_i} \tag{23}$$

where P_i and R_i represent the precision and the recall.

Precision It is the ratio of true positives and total positive class and is given by,

$$P = \frac{TP}{TP + FP} \tag{24}$$

Recall It is the ratio of the true positives to the total number of the positive class and is given by,

$$Recall = \frac{TP}{TP + FN} \tag{25}$$

4.4 Performance analysis

The performance analysis using two datasets is illustrated in this section. The analysis is performed based on accuracy, F-measure, recall, and precision by varying chunk size.

4.4.1 Analysis using Newsgroup dataset

Figure 2 illustrates the analysis of the proposed RMFO in terms of accuracy, F-measure, precision, and recall using newsgroup dataset. The analysis has been performed by varying the population size from 0 to 50. The analysis based on the accuracy parameter by varying the chunk size from two to ten is shown in Fig. 2a. When the size of chunk is 2, the corresponding accuracy parameter for RMFO with population = 10 is 89.84%, population = 20 is 90%, population = 30 is 90.12%, population = 40 is 91.332%, and population = 50 is 93.155% respectively. The performance of the proposed method has been increased when the population size increases. The analysis based on F-measure parameter with varying chunk size is depicted in Fig. 2b. For chunk size 10, the corresponding F-measure parameter for RMFO with population = 10 is 72.096%, population = 20 is 78.957%, population = 30 is 83.683%, population = 40 is 90.248%, and population = 50 is 91.615%, respectively. The analysis based on precision parameter with varying chunk size is depicted in Fig. 2c. When the size of chunk is 2, the corresponding precision parameter for RMFO with population = 10 is 86.409%, population = 20 is 91.68%, population = 30 is 91.693%, population = 40 is 92.544%, and population = 50 is 92.549%, respectively. The analysis based on recall parameter with varying chunk size is depicted in Fig. 2d. When the size of chunk is 2, the corresponding recall parameter for RMFO with population = 10 is 89.828%, population = 20 is 90.032%, population = 30 is 90.151%, population = 40 is 91.325%, and population = 50 is 93.149%, respectively.

4.4.2 Analysis using Reuter's dataset

Figure 3 illustrates the analysis of the proposed RMFO in terms of accuracy, F-measure, precision, and recall using Reuter's dataset. The analysis based on accuracy parameter by varying the chunk size from two to ten is shown in Fig. 3a. For chunk size 10, the corresponding accuracy parameter for RMFO with population = 10 is 80.823%, population = 20 is 81.727%, population = 30 is 90.5%, population = 40 is 90.8%, and population = 50 is 93.168%, respectively. The analysis based on F-measure parameter with varying chunk size is illustrated in Fig. 3b. When the size of chunk is 2, the corresponding F-measure parameter for RMFO with population = 10 is 89.797%, population = 20 is 93.833%, population = 30 is 94.214%, population = 40 is 95.069%, and population = 50 is 96%,

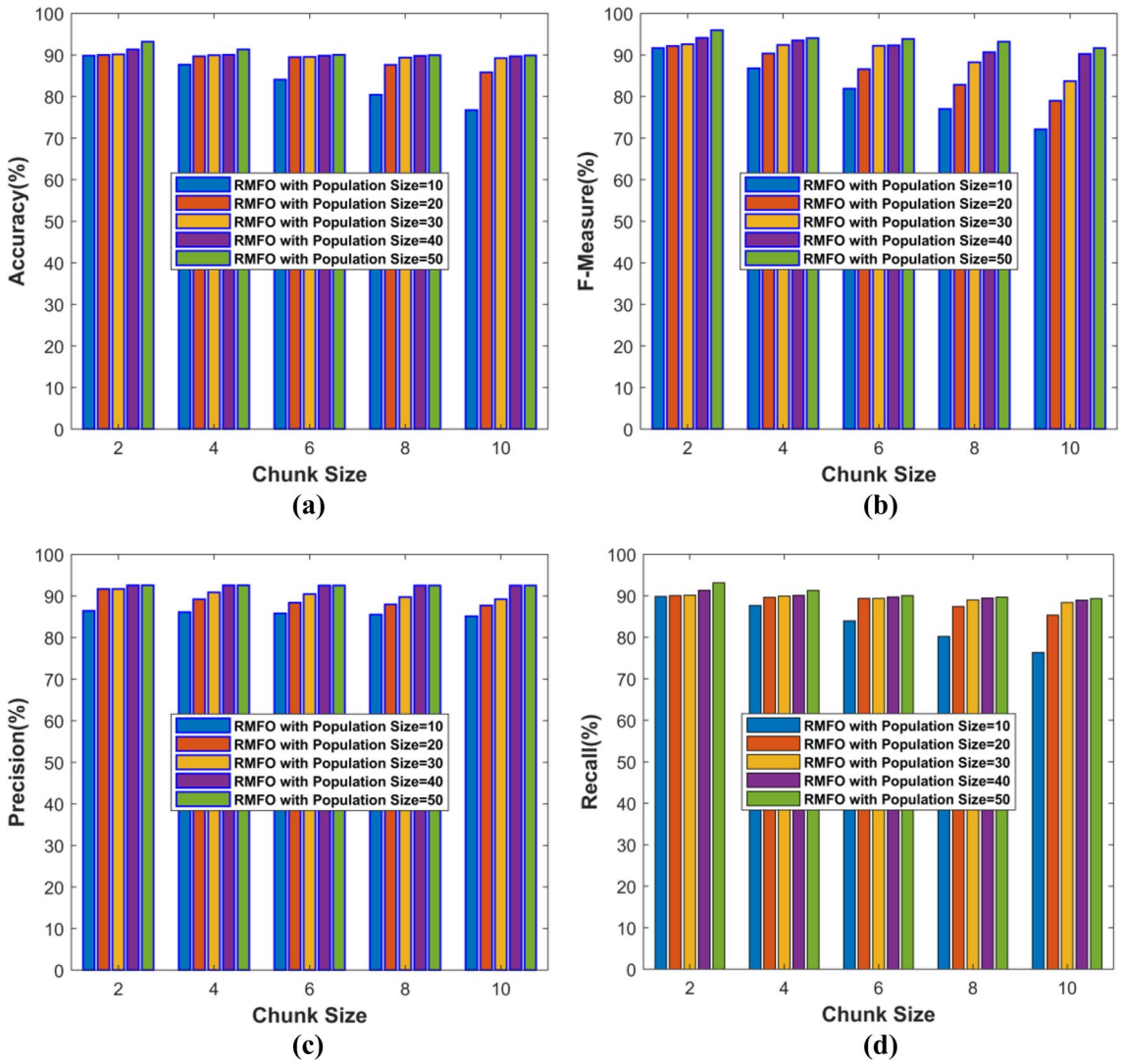

Fig. 2 Performance analysis using Newsgroup dataset in terms of **a** accuracy, **b** F-measure, **c** precision, **d** recall

respectively. The analysis based on precision parameter with varying chunk size is demonstrated in Fig. 3c. For chunk size 10, the corresponding precision parameter for RMFO with population = 10 is 80.846%, population = 20 is 88.190%, population = 30 is 91.023%, population = 40 is 91.659%, and population = 50 is 92.473% respectively. The analysis based on recall parameter with varying chunk size is shown in Fig. 3d. When the size of chunk is 2, the corresponding recall parameter for RMFO with population = 10 is 89.135%, population = 20 is 90.684%, population = 30 is 92.144%, population = 40 is 93.970%, and population = 50 is 95.25% respectively.

From Figs. 2 and 3, it can be shown that the proposed RMFO algorithm has the high performance for the maximum population size and minimum chunk size.

4.5 Comparative analysis

The comparative analysis using two datasets is illustrated in this section. The analysis is performed on the existing and proposed methodologies in terms of accuracy, F-measure, precision, and recall parameters by varying the chunk size.

4.5.1 Analysis using Newsgroup dataset

Figure 4 elaborates the comparative analysis of proposed RMFO with existing WPLSA, ICGT, and CBICA in terms of accuracy, F-measure, precision, and recall using newsgroup dataset. The analysis based on the accuracy parameter by varying the chunk size from two to ten is shown in Fig. 4a. When the size of chunk is 2, the corresponding accuracy

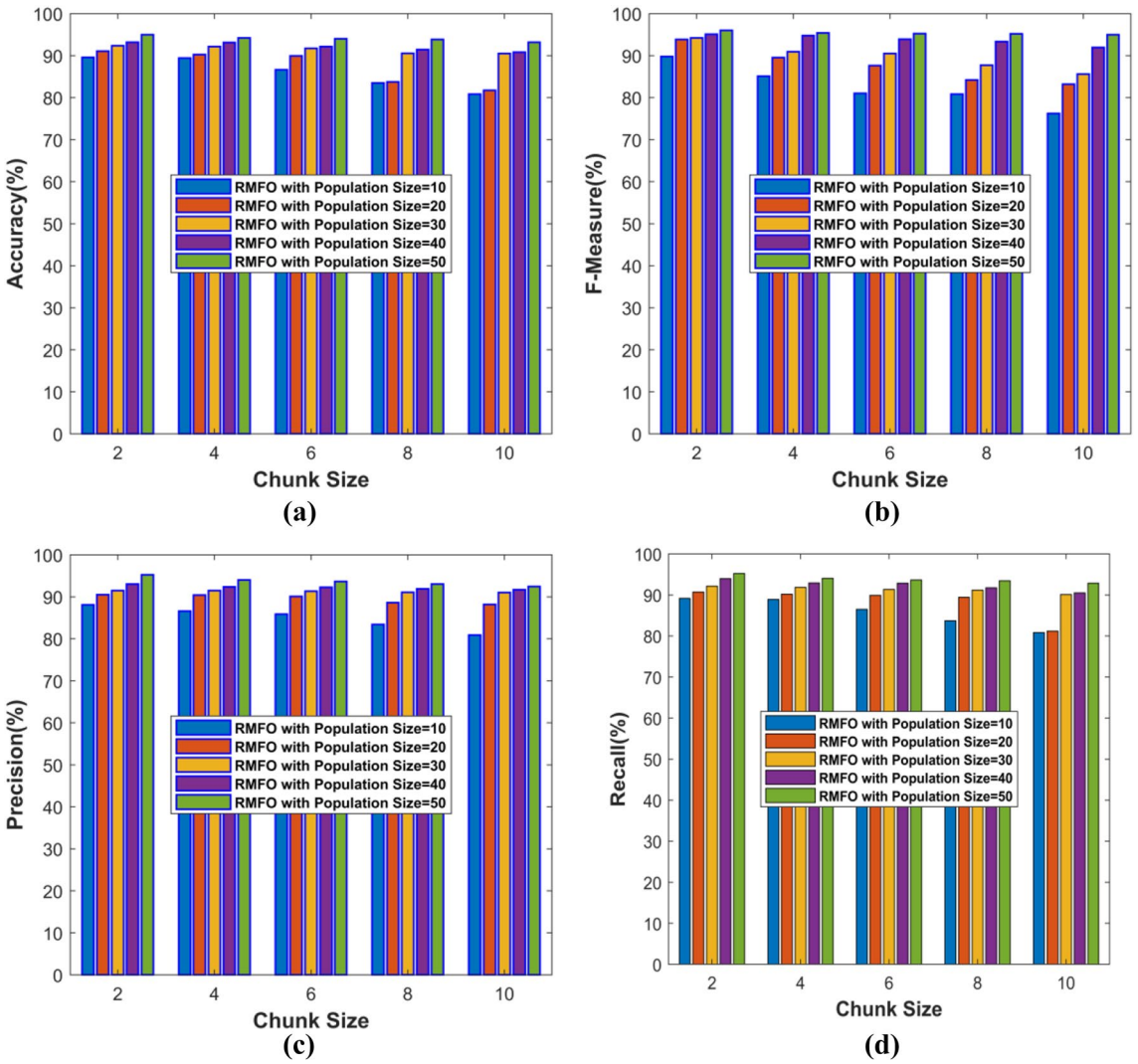

Fig. 3 Performance analysis using Reuters dataset in terms of **a** accuracy, **b** F-measure, **c** precision, **d** recall

parameter measured by the proposed algorithm is 93.98%, whereas the accuracy values of the existing WPLSA, ICGT, and CBICA are 93.457%, 92.918%, and 90.06%, respectively. Here, the performance of the proposed RMFO algorithm is 0.56%, 1.13%, and 4.17% higher than the performance of the existing methods, such as WPLSA, ICGT, and CBICA. The analysis based on F-measure parameter with varying chunk size is illustrated in Fig. 4b. When the size of chunk is 2, the corresponding F-measure parameter computed by existing methods, such as WPLSA, ICGT, and CBICA and the proposed RMFO are 94.801%, 93.976%, 92.269% and 94.875%, respectively. Here, the best performance has been achieved by the proposed method while the worst has been achieved by the CBICA.

The analysis based on precision parameter with varying chunk size is demonstrated in Fig. 4c. When the size of chunk is 2, the corresponding precision values computed by existing WPLSA, ICGT, and CBICA and the proposed RMFO are 92.5198%, 60%, 60%, and 93.958%, respectively. Here also, the proposed method shows the best performance than the existing methods by having the high precision value. Among the existing methods, WPLSA has the best performance. However, the performance of WPLSA is 1.53% lower than the performance of the proposed RMFO algorithm. The analysis based on recall parameter with varying chunk size is shown in Fig. 4d. For chunk size 10, the corresponding recall values computed by existing WPLSA, ICGT, and CBICA and the proposed RMFO are 91.135%, 90.436%, 76.371%, and 92.064%, respectively. From Fig. 4, it is exposed that the proposed RMFO has the maximum performance by obtaining the high values for accuracy, F-measure, precision, and recall.

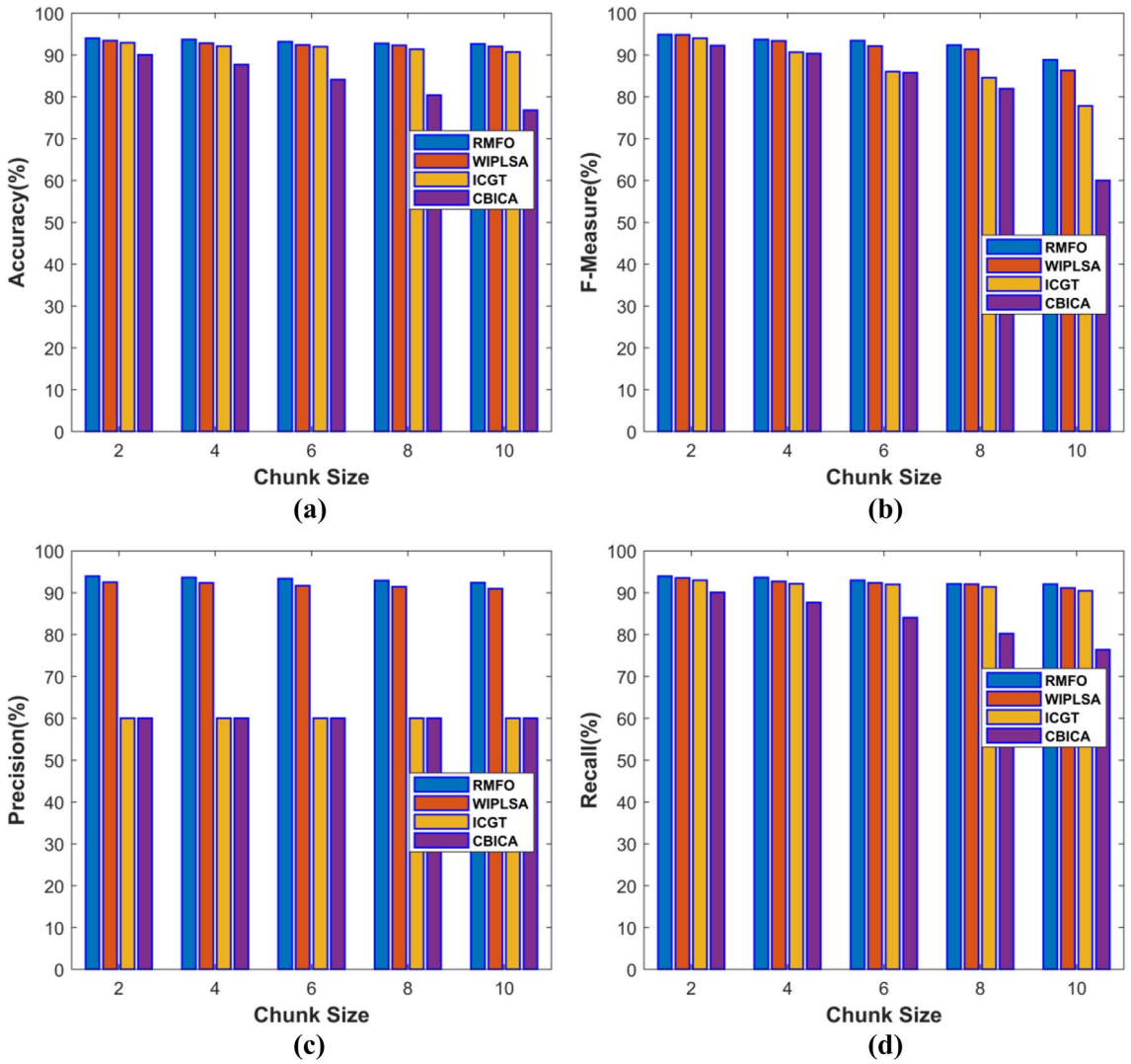

Fig. 4 Comparative analysis using Newsgroup dataset in terms of **a** accuracy, **b** F-measure, **c** precision, **d** recall

4.5.2 Analysis using Reuter's dataset

Figure 5 elaborates the comparative analysis of proposed RMFO with existing WPLSA, ICGT, and CBICA in terms of accuracy, F-measure, precision, and recall using Reuters dataset. The analysis based on accuracy parameter by varying the chunk size from two to ten is shown in Fig. 5a. When the size of chunk is 2, the accuracy value obtained by the proposed method is 93.407%, which is 0.33%, 0.7%, and 2.24% higher than the accuracy of the existing methods, such as WPLSA, ICGT, and CBICA, respectively. The analysis based on F-measure parameter with varying chunk size is illustrated in Fig. 5b. When the size of chunk is 2, the corresponding F-measure parameter computed by existing WPLSA, ICGT, and CBICA and proposed RMFO are 93.407%, 92.994%, 87.210%, and 93.645%, respectively. Likewise, for chunk size 10, the corresponding F-measure parameter computed by existing WPLSA, ICGT, and CBICA and proposed RMFO are 91.884%, 90.226%, 60% and 92.132%, respectively. When the chunk size increases the performance of the comparative methods decreases. However, the proposed algorithm has the high performance than the existing method by obtaining the maximum F-measure.

The analysis based on precision parameter with varying chunk size is demonstrated in Fig. 5c. When the size of chunk is 2, the corresponding precision values computed by existing WPLSA, ICGT, and CBICA and proposed RMFO are 92.626%, 60%, 60%, and 93.649%, respectively. When the chunk size is 2, the proposed method shows 1.09% improvement than the WPLSA and 35.93% improvement than ICGT and CBICA. The analysis based on recall parameter with varying chunk size is shown in Fig. 5d. When the size of chunk is 2, the corresponding recall values computed by existing WPLSA, ICGT, CBICA, and proposed RMFO

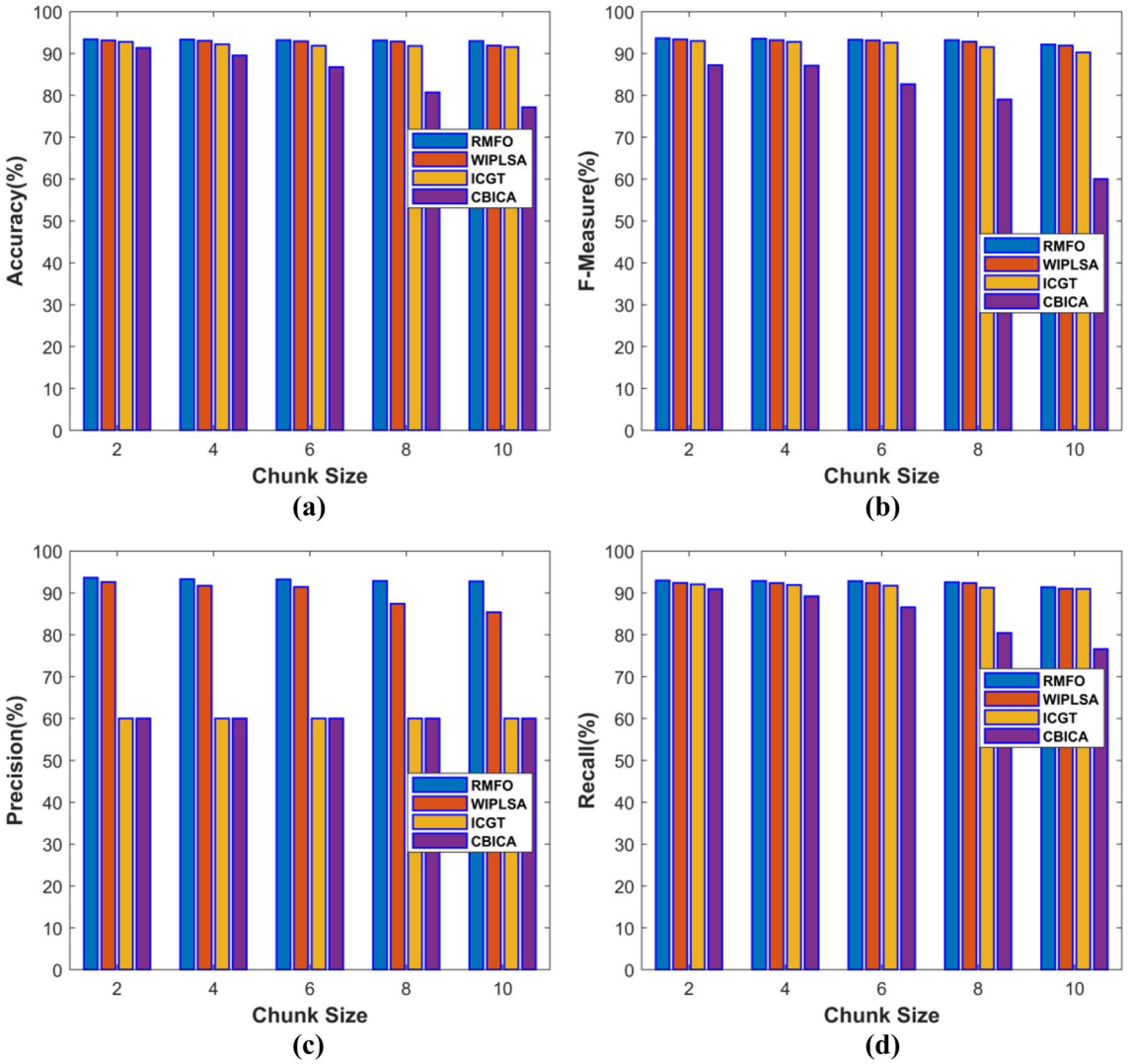

Fig. 5 Comparative analysis using Reuters dataset in terms of **a** accuracy, **b** F-measure, **c** precision, **d** recall

are 92.392%, 92.041%, 90.919%, and 92.949% respectively. Likewise, for chunk size 10, the corresponding recall values computed by the existing methods, such as WPLSA, ICGT, and CBICA, the proposed RMFO are 91.031%, 90.961%, 76.581%, and 91.372%, respectively. From Fig. 5, it can be seen that the proposed RMFO algorithm has the high performance than the existing methods.

Table 1 Comparative discussion

Metrics	WPLSA	ICGT	CBICA	Proposed RMFO
Accuracy (%)	93.458	92.919	91.317	93.98
F-measure (%)	94.802	93.976	92.269	94.876
Precision (%)	92.626	60	60	93.958
Recall (%)	93.525	92.994	90.919	93.964

4.6 Comparative discussion

Table 1 depicts the comparative discussion of the existing WPLSA, ICGT, CBICA, and proposed RMFO in terms of accuracy, F-measure, precision, and recall parameters. Here, the Table has been filled with the maximum performance obtained by the comparative methods. The maximum performance measured by the proposed RMFO in terms of accuracy parameter is 93.98%, whereas the accuracy values of existing WPLSA, ICGT, and CBICA are 93.458%, 92.919%, and 91.317%, respectively. Here, the accuracy of the proposed method is 0.56%, 1.13%, and 2.83% higher than the accuracy of the existing methods, such as WPLSA, ICGT, and CBICA, respectively. Similarly, the proposed RMFO has the maximum F-measure, precision, and recall than the exiting methods.

When compared to the existing methods, such as WPLSA, ICGT, and CBICA, the proposed RMFO algorithm offers high performance by obtaining the maximum accuracy, F-measure, precision, and recall values. One of the reasons for the high performance of the proposed method is that it has the advantages of both ROA and MFO. The ROA is highly efficient and performs effectively in the fictional computing for solving the optimization issues. MFO offers better convergence rate, while obtaining global optimal solution. It poses the ability to balance the exploration and exploitation in a proper manner. Another reason is that, the use of the fuzzy bounding model for setting the boundary threshold to place the document in a specific cluster. The concept of fuzzy set promises a more appropriate representation of the decision processes. Also, the fuzzy decision making processes the uncertainty more accurately.

5 Conclusion

This paper presents an effective incremental document clustering method using the hybrid fuzzy bounding degree. The proposed model undergoes three steps for the document clustering, namely Pre-processing, feature extraction, and Incremental document categorization. The pre-processing step is a significant step for the effective management of the proposed incremental document categorization, which enable the research to concentrate on the document clustering of the dynamic data. The processes associated with the pre-processing step are stop-word removal and stemming. The second step is the feature extraction using the TD-IDF and Wordnet features. Finally, the incremental document clustering is performed based on the hybrid fuzzy bounding degree that includes the RMFO and boundary degree, RMFO aims to select the optimal weights for the boundary degree model. The RMFO is the combination of ROA and MFO. The experimentation is performed using two databases, namely 20 newsgroup databases and the Reuter database. In addition, the proposed RMFO outperformed other existing methods based on accuracy, F-measure, precision, and recall with maximal values of 93.98%, 94.876%, 93.958%, and 93.964%, respectively. The future dimension of the research will be based on the enhanced algorithms for document retrieval mechanisms, which would render a higher retrieval accuracy.

References

1. Chevalier M, El Malki M, Kopliku A, Teste O, Tournier R (2016) Implementation of multidimensional databases with document-oriented NoSQL. In: Big data analytics and knowledge discovery, pp 379–390
2. Martinho B, Santos MY (2016) An architecture for data warehousing in big data environments. In: Research and practical issues of enterprise information systems, vol 268, pp 237–250
3. Doermann D (1998) The indexing and retrieval of document images: a survey. Comput Vis Image Underst 70(3):287–298
4. Callan JP (1994) Passage-level evidence in document retrieval. In: SIGIR. Springer, Berlin, pp 302–310
5. Hao S, Shi C, Niu Z, Cao L (2018) Concept coupling learning for improving concept lattice-based document retrieval. Eng Appl Artif Intell 69:65–75
6. Mothe J, Chrisment C, Dousset B, Alaux J (2003) DocCube: multi-dimensional visualisation and exploration of large document sets. J Am Soc Inf Sci Technol 54(7):650–659
7. Slonim N, Tishby N (2000) Document clustering using word clusters via the information bottleneck method. In: Proceedings of the 23rd annual international conference on research and development in information retrieval, pp 208–215
8. Karypis MSG, Kumar V, Steinbach M (2000) A comparison of document clustering techniques. In: Proceedings of TextMining workshop at KDD2000, May 2000
9. Li N, Luo W, Yang K, Zhuang F, He Q, Shi Z (2018) Self-organizing weighted incremental probabilistic latent semantic analysis. Int J Mach Learn Cybern 9(12):1987–1998
10. Wan Y, Liu X, Wu Y, Guo L, Chen Q, Wang M (2018) ICGT: a novel incremental clustering approach based on GMM tree. Data Knowl Eng 117:71–86
11. Sangaiah AK, Fakhry AE, Abdel-Basset M, El-Henawy I (2018) Arabic text clustering using improved clustering algorithms with dimensionality reduction. Cluster Comput 22:1–15
12. Kotte VK, Rajavelu S, Rajsingh EB (2019) A similarity function for feature pattern clustering and high dimensional text document classification. Found Sci. https://doi.org/10.1007/s10699-019-09592-w
13. Mulay P, Shinde K (2019) Personalized diabetes analysis using correlation-based incremental clustering algorithm. In: Mittal M, Balas VE, Goyal LM, Kumar R (eds) Big data processing using spark in cloud. Springer, Berlin, pp 167–193
14. Madhusudhanan S, Jaganathan S (2018) Incremental learning for classification of unstructured data using extreme learning machine. Algorithms 11(10):158
15. Kannan J, Shanavas AM, Swaminathan S (2018) SportsBuzzer: detecting events at real time in Twitter using incremental clustering. Trans Mach Learn Artif Intell 6(1):01
16. Liu Y, Chen J, Wu S, Liu Z, Chao H (2018) Incremental fuzzy C medoids clustering of time series data using dynamic time warping distance. PLoS ONE 13(5):0197499
17. Binu D, Kariyappa BS (2018) RideNN: a new rider optimization algorithm-based neural network for fault diagnosis in analog circuits. IEEE Trans Instrum Meas 68:2–26
18. Mirjalili S (2015) Moth–flame optimization algorithm: a novel nature-inspired heuristic paradigm. Knowl Based Syst 89:228–249
19. Sedding J, Kazakov D (2004) WordNet-based text document clustering. In: Proceedings of the 3rd workshop on robust methods in analysis of natural language data, pp 104–113
20. Yarlagadda M, Gangadhara Roa K, Srikrishna A (2019) Frequent itemset-based feature selection and Rider Moth Search Algorithm for document clustering. J King Saud Univ Comput Inf Sci. https://doi.org/10.1016/j.jksuci.2019.09.002
21. Xu Z, Xia M (2011) Distance and similarity measures for hesitant fuzzy sets. Inf Sci 181(11):2128–2138
22. Newsgroup database. http://qwone.com/~jason/20Newsgroups/. Accessed Oct 2018
23. Reuter Database. https://archive.ics.uci.edu/ml/datasets/reuters-21578+text+categorization+collection. Accessed Oct 2018