Svedberg Open
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

An Optimized Document Information Retrieval Framework Using Clustering Techniques Integrated with Bacterial Foraging Optimization

S.Nancy Lima Christy[1], Dr.A.Jeeva[2], Nisha Boopathy[3], Ali Bostani[4], Dr.A.Mummoorthy[5], Suresh Arumugam[6], Dr .P.DharmendraKumar[7]

[1]Assistant Professor, Department Of Computer science and Engineering, SRM Institute of Science and Technology, Ramapuram Campus,Chennai, Email: nancylis@srmist.edu.in, https://orcid.org/0009-0003-2514-7412

[2]Assistant Professor, Department of Mathematics, Vel Tech Rangarajan Dr.sagunthala R&D Institute of Science and Technology, Avadi, Chennai-600062. Email id:drjeevaa@veltech.edu.in, Orcid: 0000-0003-4087-2041.

[3]Associate Professor, Community Medicine, Saveetha Medical College, Saveetha Institute of Medical and Technical Sciences, Chennai, Tamilnadu, India. Email: nisha.utsav20@gmail.com

[4]Associate Professor, College of Engineering and Applied Sciences, American University of Kuwait, Salmiya, Kuwait. Email: abostani@auk.edu.kw, 0000-0002-7922-9857

[5]Professor, Department of Computer Science and Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, Email: drmummoorthya@veltech.edu.in, ORCID ID: https://orcid.org/0000-0002-1820-2124

[6]Scientist, Central Research Laboratory, Meenakshi Medical College Hospital & Research Institute, Meenakshi Academy of Higher Education and Research, Chennai, tamilnadu, India. Email: Suresh@maher.ac.in

[7]Assistant professor, Department of CSE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, AP, India, Email: pdharmendrakumar@kluniversity.in, dharmendra.phd.au@gmail.com, https://orcid.org/0009-0006-5087-9123

Abstract

In this study, a new Document Information Retrieval (DIR) framework is designed using K-means clustering method with Bacterial Foraging Algorithm which will overcome the scalability and cost computation required while adaptation with large document. The documents are first pre-processed using vector space representation with TF-IDF weighting and partitioned into clusters to reduce the search space dimensionality. Followed by this BFA used for smart doc exploration which takes the support of chemotaxis, swarming, reproduction, and elimination-dispersal to find the related doc based on the user query. Integrating clustering with swarm intelligence can improve retrieval accuracy and reduce redundancy and time. The DIR-BFA model outperforms the existing approaches (EQS and Firefly-based retrievals) in the performance evaluation based on standard metrics (precision, recall and f-measure). Experiment results showed that the retrieval accuracy was improved and the run time was reduced effectively.

Keywords: Cluster, swarm intelligence, centroid, information retrieval, and chemotaxis.

This is an open access article under CC BY 4.0, allowing unrestricted use with proper attribution, a license link, and indication of any changes made.

1. Introduction

Information retrieval is the progression of obtaining needed information from the huge collection of resources whereas, document information retrieval is acquiring needed information from the document or retrieving the document from the collection of documents. The information retrieval progression is initiated when the user enters the user query into the system [1]. The necessary information is framed as a formal query statement for example, in the search engine search strings are considered as a query.A single object in the source of the

document is not spotted with the query and the query may match various objects, perhapswith a diversified degree of relevancy [2].

Data mining based techniques incorporated in the pre-processing phase of the IR process from the collection of documents. The knowledge acquired from the pre-processing phase is used in the subsequent document exploration process. The information extracted from pre-processing is fragmented into numerous groups [4].Let the collection of the document is represented as D={d1,d2….d_n}be the collection of n documents and K={k1,k2…km} is the collection of m terms. Every document is comprised of a subset of the terms that are composed in K. The user request REQ is denoted by the set of terms. The document information retrieval is the progression of identifying appropriate documents in relevance to the user REQ from the collection of sources.The DIR scans every document and estimates the score among every individual document and the given request of the user. The relevant documents and the information are retained using the ranking function. The available DIR process has numerous polynomial complexity. Document Information Retrieval has polynomial complications and the run time is exceptionally high when processing the huge document.Hence, DIR is incorporated with the Bacterial Foraging Algorithm that uses k-means clustering for pre-processing.

The occurrence of bacteria can be spotted everywhere from the hostile atmosphere to the hospitable surrounding. Bacteria are identified in diverse nature, multifarious functional character, and also has diverse behavioral patterns. Based on the behavior of bacteria an algorithm is developed to obtain the solution for the optimization problem [4].Bacterial Foraging Algorithm (BFA) is formulated from the foraging strategy of E-Coli.The dispersal of an element is eliminated by adopting the BFA process whereas the process improves the performance of the information retrieval. The document exploration process is also effective and the irrelevancy is eliminated at the pre-processing stage itself.

The optimization problem reflects the collection of the document as a solution space in the huge resource. The diversification and intensification approaches permit the identification of a needed subset of information from a document with a minimal computational time.Moreover, the communication among the document and the query is achieved by the bacteria that permits the identificationof the high-quality solutions associatedwith the evolutionary ones. Nevertheless, the process of identification of documents is still stochastic, and when the solution space is huge, the bacteriain the formulated space aredisorientated, which minimizes the excellence of the final subset of documents reverted by the bacteria.

Motivated by the achievement of BFA in dealing with numerous optimization issues, themain intent of this paperis to design and inspect a new bacterial foraging algorithm for formulating the solution for the DIR problem. The proposed algorithm explores the information retrieved in the pre-processing step by incorporating the clustering technique that guides the bacteria in the exploration of the space of documents.

The remainder of the paper is emphasized as follows, the various similar existing optimization and data mining approach for information retrieval is explained in Section 2, the proposed methodology and the system design is explained in Section 3, experimental results are illustrated in section 4 and the DIR-BFA is concluded with future work suggestion in Section 5.

2. Literature Review

In order to retrieve information from a document or needed document from the group of the resource is attained by various researches and the researchers. In this section, data mining and bio-inspired document information retrieval are explained.

Association Rule Mining for Information Retrieval [5] is developed for text processing and this approach is modeled by the given request as well as with the set of concepts. Whereas, the correlation among the concepts of similar requests is determined by the process of Association Rule Mining.Pattern Taxonomy Mining (PTM) [6]is projected to improve the performance of the needed documents. The conception of user request is used in the pattern taxonomy mining and it is identified by incorporating the closed algorithm relies on the documents in the training set. PTM minimizes the noise among the set of terms and user request in the collection of documents.

Lazy Associative Tag REcommendor [7] is proposed to retrieve the association rules from the set of documents in the training phase and K-Nearest Neighbors for Information Retrieval [8] is projected to acquire the document using the weighting approach. Whereas, it uses the support vector model. The ranking function is used to arrange the documents in the required order [9] and the generated rule gives the score for the document. The retrieval process is complicated when the size of the document or resource is huge.

Firefly [10] algorithm is developed to retrieve the document in the context of medical and the process of retrieving is effective. Genetic Algorithm for Feature Selection [11] combines the stochastic and feature election that deals with the retrieval of the information process. Evolutionary Query Sampling [12] approach is established to acquire the huge documents and the matching mechanism is implemented among the user query and the document source. Dynamic economic dispatch [13] and Numerical optimization approach [14] are successfully incorporated with the Bee Swarm Optimization (BSO). The optimization algorithms have shown promising outcomes and the effectiveness is reduced when the document source is large.

3. Proposed Framework

Swarm intelligence is developed based on the cumulative behavior of self-governed and decentralized systems. Swarm intelligence is a prominent optimization approach. The process of modeling problem solving and searching system with the optimization approach is effective. The proposed DIR-BFA has been formulated from the biological pattern and the behavior of bacteria. The BFA is designed using the E-coli bacteria and the mechanism is characterized as Chemotaxis, swarming reproduction, and elimination of dispersal events. The proposed Document Information Retrieval with Bacterial Foraging Algorithm (DIR-BFA) is composed of two stages namely pre-processing and document exploration.

3.1.Pre-Processing

In the first stage of pre-processing, document decomposition is carried and the k means clustering is applied to decompose the document without compromising the generality of the document. The k-means algorithm is generally represented as follows

$$De = \sum_{j=1}^{K} \sum_{m \in S_j} |x_n - \mu_j|^2$$

where the n^{th} data point and the vector is represented by X_n. μ_j represent the data point of the centroid in S_j.

The main intent of K-means is clustering or partitioning n number of data points into the k (cluster count) disjoint subset of S_j. Every subset $S_j$is comprised of n_j data points. The criterion of total square value is minimized that is the main aim of this algorithm. The k number of clusters is assigned with a random number of data points and the centroid of every cluster is estimated. The acquired point is allocated to the obtained cluster where the centroid value is closest to that point. The estimation process is repeated until a new data point is assigned to the cluster. The k-means for the DIR is denoted in the following section.

3.1.1. Document Representation

The vector space model is incorporated to signify the document space model. Every document DoC in the source is denoted by a vector $\{v_1,v_2,.....v_n\}$ where the weight of the term te is denoted by w_i. The importance of every term in the document is represented by the weight value of the term. The term weight is computed using the Term Frequency with Inverse Document Frequency (TF-DF). The estimation of TF-DF is equated as,

$$v_{ij} = to_{ji} \times ido_{ji}$$

where v_{ij} denotes the weight of the term i in the j^{th} document.to_{ji} is the count of the incidence of the term i in the document j. The ido_{ji} value is equated as,

$$ido_{ji} = \log\left(\frac{n}{do_{ji}}\right)$$

where do_{ji} signifies the frequency in the collection of n documents.

3.1.2. Similarity Computation

The measure of similarity among the DoC_i and DoC_j is estimated using the cosine correlation value is estimated by

$$\cos(DoC_i, DoC_j) = \frac{DoC_i^t DoC_j}{|DoC_i||DoC_j|}$$

where $DoC_i^t DoC_j$ indicates the dot product of the two individual documents vectors DoC_i *and* DoC_j. $|DoC_i|$ represents the length of the vector DoC_i that is the count of the terms holding weights with a null value in the documents DoC_i.

3.1.3. Centroid Updating

The updation of the centroid is estimated as,

$$G_i = \frac{1}{|CL_i|}\sum_{j=1}^{|C_i|} DoC_j$$

where G_i represents the new center of the cluster CL_i.

The Closed Frequent Pattern Mining is applied to the clustered document where the clustering is attained by the k-means clustering. The Closed Frequent Pattern Mining returns the set of frequent items. The overall system design is illustrated in Figure1.

System Design

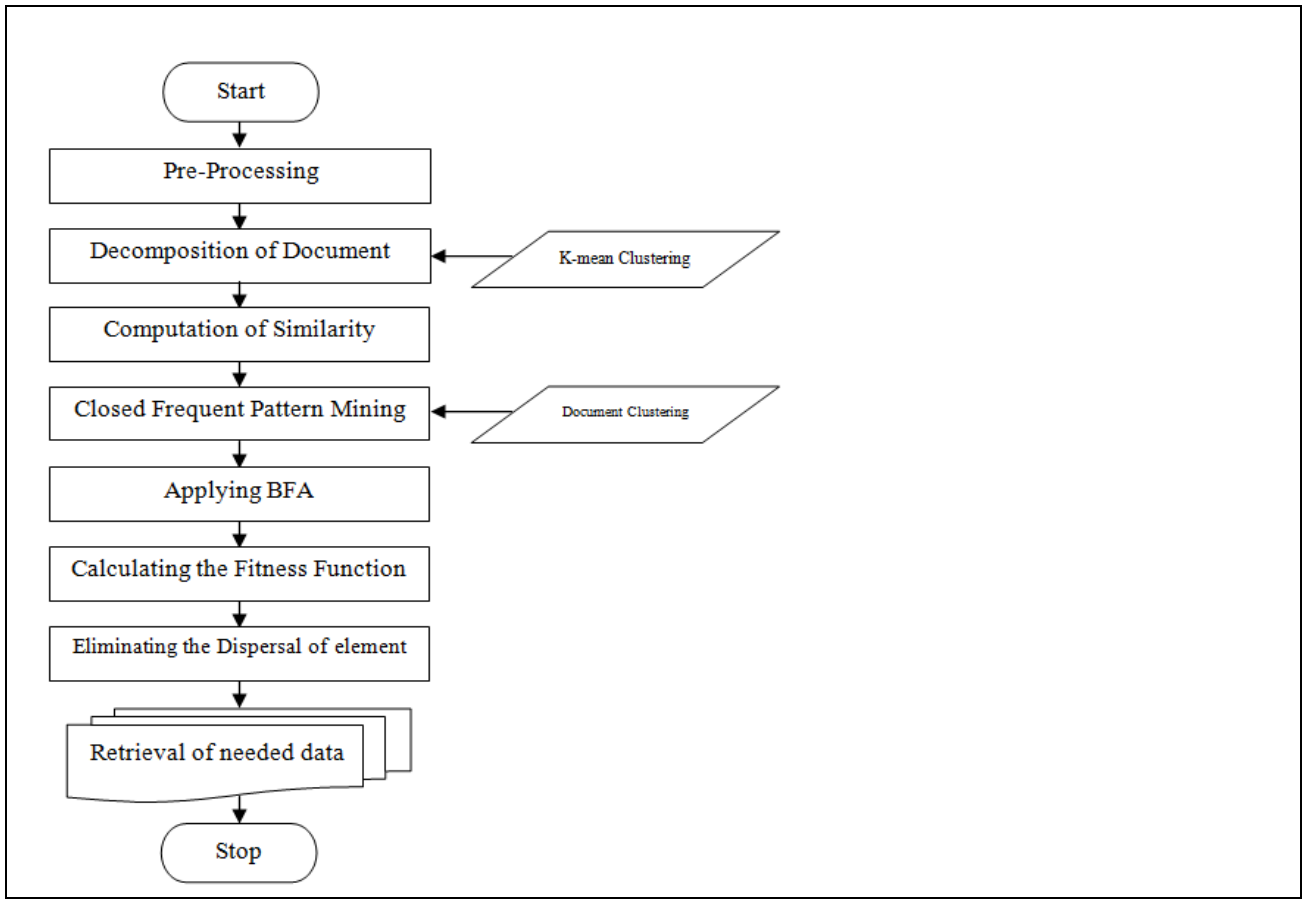

Figure 1: Overall system design of DIR-BFA

3.2. Document Information Retrieval with Bacterial Foraging Algorithm (DIR-BFA)

According to BFA, a matching procedure is applied to every cluster that resides inside the document, and the request of the user. This procedure used in BFA returns the count of the common terms among the document subset and the request. The best solution identified by every bacteria is saved in the foraging table. At the end of the procedure, the acquired best solution is transmitted to the next iteration. The process of the algorithm is repeated until it reaches the termination criteria. The process of exploration process is explained as follows,

3.2.1. Chemotaxis (Matching)

The E.Coli bacteria is accomplished via swimming and trembling whereas searching for documents in a similar direction and random direction respectively. The unwanted terms are removed and the bacterium moves along the entire document to search for the relevant term. To match the term, the matching operation is established that computes the n_i value of every cluster C_i relevant to the assigned request REQ as follows:

$$Chemotaxis(REQ, CL_i) = n_i = \left| \bigcup_{i=0}^{CF_i} (REQ \cap CF_i^j) \right|$$

where CF_i^j is the jth set of terms in the closed frequent pattern CF of the cluster C_i. The probability factor of electing a cluster C_i is denoted by

$$PB_i = \frac{n_i}{n}$$

For a instance, consider ten documents DOC = { DoC_1, DoC_2 . . . DoC_{10} }, five required terms {te_1, te_2,te_3 . . . te_5 } , and three clusters { CL_1, CL_2 ,CL_3 } represented as follows:

CL_1 = { DoC_1 ,DoC_2 ,DoC_3 } with the closed frequent patterns of items: CF_1 ={{te_1,te_2},{te_1,te_3}}.

CL_2 = { Doc_4 ,DoC_5 ,DoC_6 } with the closed frequent patterns of items: CF_2 ={{te_2},{te_4}}.

CL_3 = { DoC_7 ,DoC_8 ,DoC_9 ,DoC_{10} } with the closed frequent patterns of items: CF_3 ={{te_4,te_5},{te_3,te5}} .

If we have the following request REQ = {te_1,te_2,te_3}, the matching function between each cluster and REQ is determined as follows:

n_1 = | (CF^1_1 ∩ REQ) ∪ (CF^2_1 ∩ REQ) | = |{te_1,te_2,te_3}| = 3 .

n_2 = | (CF^1_2 ∩ REQ) ∪ (CF^2_2 ∩ REQ) | = |{te_2}| = 1 .

n_3 = | (CF^1_3 ∩ REQ) ∪ (CF^2_3 ∩ REQ) | = |{te_3}| = 1 .

Furthermore, the probability of every document in DOC is estimated as follows:

PB_1 = 3 / 5 = 60% (Probability of the cluster CL_1 is the probabilities of documents DoC_1, DoC_2 and DoC_3.)

PB_2 = 20% (Probability of the cluster CL_2 is the probabilities of documents DoC_4 ,DoC_5 and DoC_6 .)
PB_3 = 20% (Probability of the cluster CL_3 is the probabilities of documents DoC_7, DoC_8 ,DoC_9 and DoC_{10}.).

3.2.2. Swarming

Every solution is a vector ve of l number of elements where every element S_i is an integer and the value ranges from the initial value 1 to m. The document solution and the relevant term is encoded in the swarming phase. The permissible solutions in the solution space are represented by the documents.

3.2.3. Reproduction and the Fitness estimation

The estimation of fitness is according to the REQ is decided by accepting the Jaccard coefficient:

$$Fit_{max}(S) = \sum_{i=1}^{|S|} \frac{|REQ \cap DoC_{s[i]}|}{|REQ| \times |DoC_{s[i]}|}$$

3.2.4. Elimination of dispersal events

The solution obtained from the previous phase is passed to the elimination phase and every bacteria is updated to the solution space by incorporating the probability vector. The iterations with a similar element are discarded at the elimination process. The best solution acquire at the end of the process.

4. Performance Evaluation

The testing of the proposed DIR-BFA framework is carried out on a heterogeneous corpus consisting of web documents and wiki-based textual resources. These web documents and textual resources ensure diversity across document structure, term distribution, and semantics of the documents. The terms in the Collection are weighted using TF-IDF. Each document is represented using a vector space model. The dataset undergoes tokenization, stop-word removal and clustering through K-means which reduces dimensionality and partitions document space. The effectiveness of retrieval is analyzed via standard Information Retrieval (IR) performance metrics (precision, recall and F-measure). Precision refers to the number of relevant documents retrieved over the total number of documents retrieved. Formally it is defined as the ratio of true positives to true positives plus false positives. In other words, precision indicates the exactness of the retrieval.Recall measures the ability of the system to retrieve all relevant documents, computed as the ratio of true positives to the sum of true positives and false negatives, indicating completeness. F-measure, the harmonic mean of precision and recall, provides a balanced evaluation by integrating both metrics, thereby addressing trade-offs between accuracy and coverage in large-scale document retrieval scenarios.

This study evaluates the proposed Document Information Retrieval using the Bat-inspired Firefly Algorithm (DIRBFA) against two established baselines—EQS and FDIR—across five experimental dimensions: cluster count, minimum support threshold, document volume, user request load, and iterative convergence. Performance is measured via F-measure, CPU runtime, and document quality. Across all scenarios, DIRBFA consistently delivers superior retrieval accuracy while maintaining competitive or lower execution times.

Table 1: Document Quality (thousands)

Docs (K)	EQS	FDIR	DIRBFA
100	0.24	0.34	0.36
200	0.25	0.30	0.33
500	0.28	0.24	0.29
800	0.15	0.21	0.24
1200	0.13	0.19	0.22

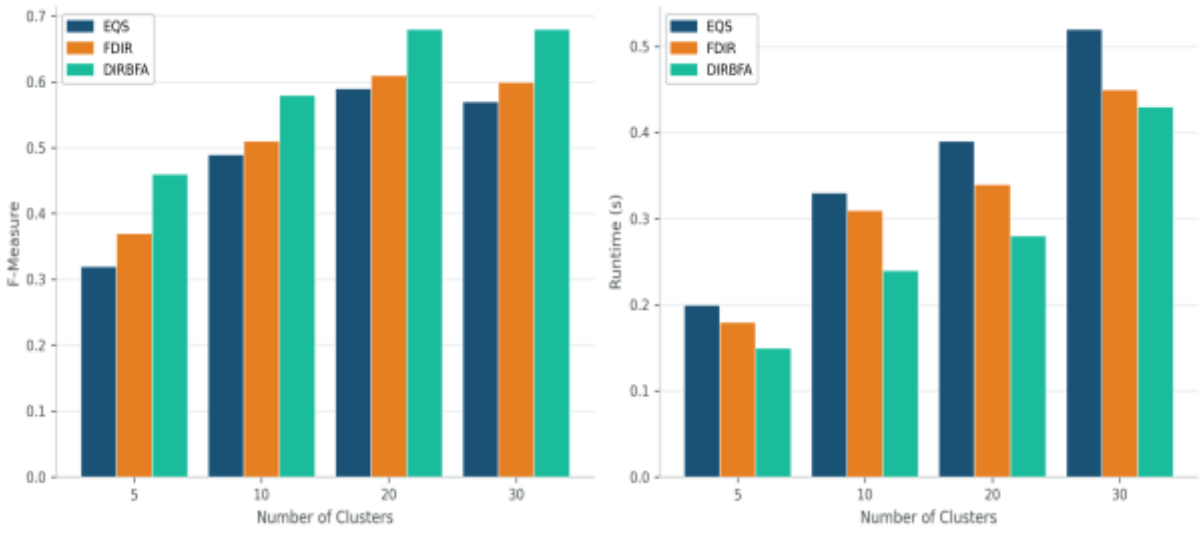

Figure 1: F-Measure & Runtime vs. Number of Clusters

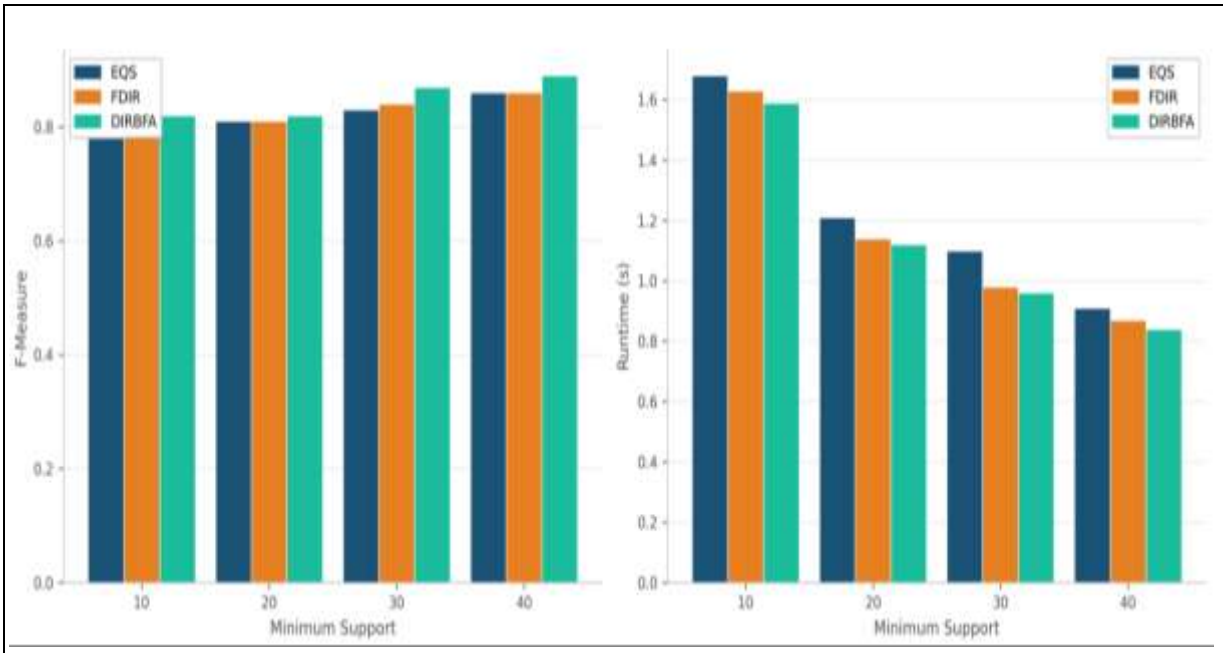

Figure 2: F-Measure & Runtime vs. Minimum Support

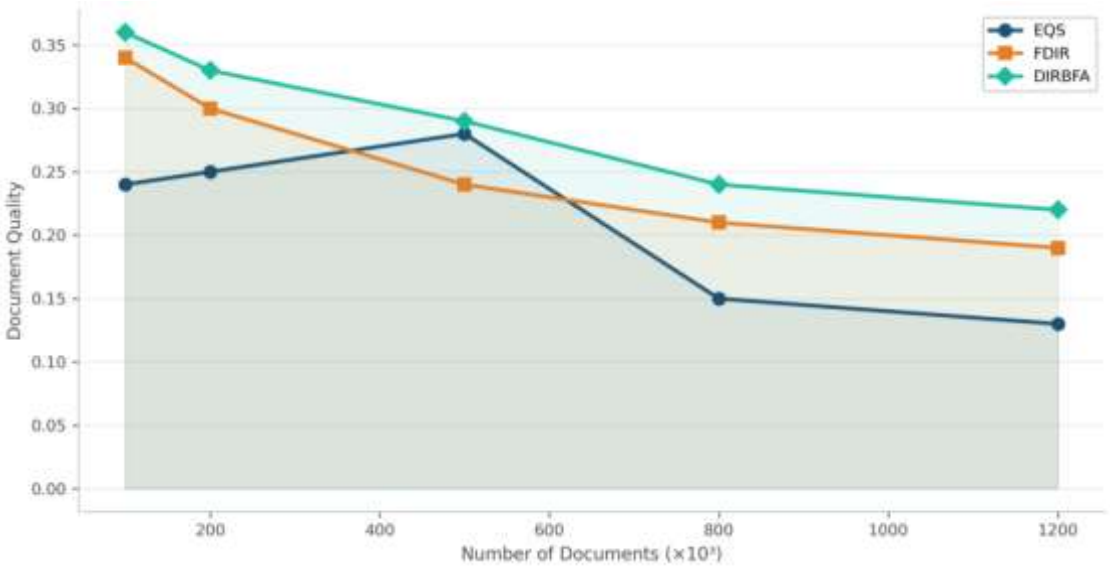

Figure 3: Document Quality vs. Document Volume

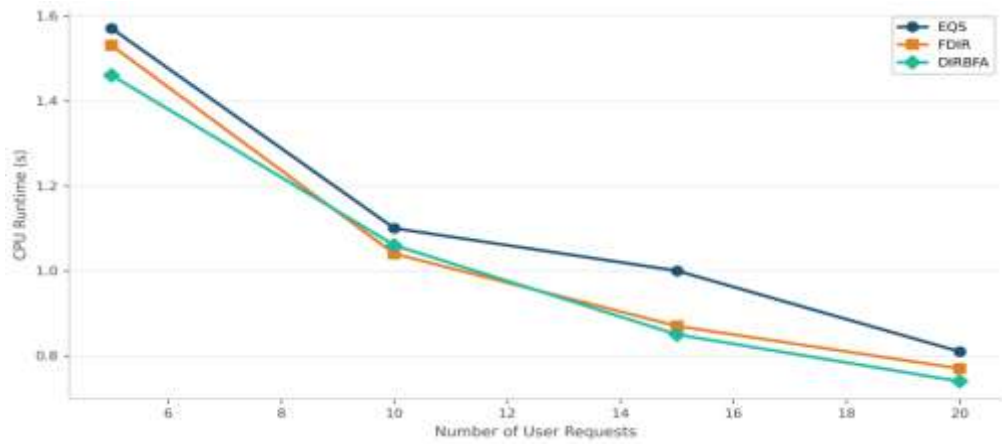

Figure 4: CPU Runtime vs. User Requests

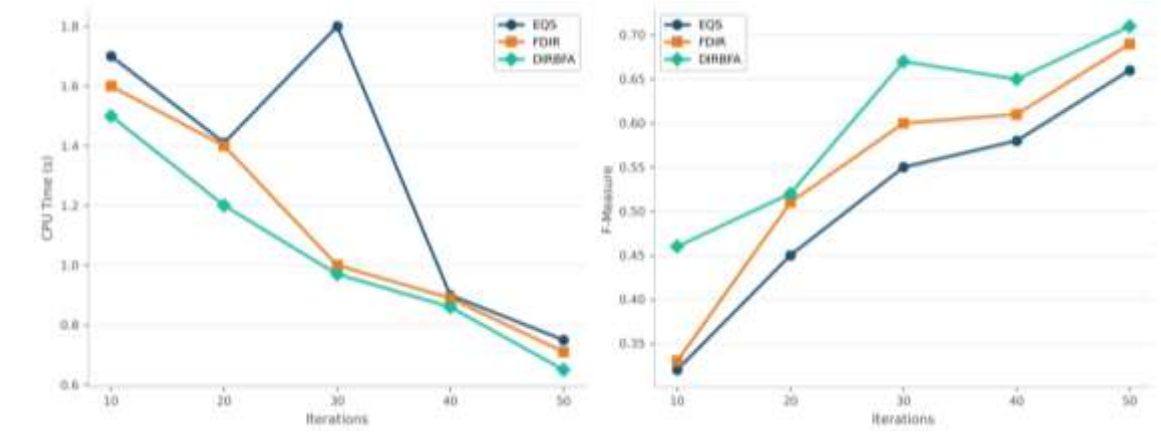

Figure 5: CPU Time and F-Measure vs. Iterations

When the number of clusters was varied (5–30), DIRBFA achieved the highest F-measure at every level—peaking at **0.68** for clusters 20 and 30, compared to 0.61 and 0.59 for FDIR and EQS respectively. Runtime also

favoured DIRBFA, recording 0.15 s at 5 clusters against EQS's 0.20 s. Similarly, under varying minimum support (10–40), DIRBFA's F-measure reached **0.89** at support=40, outperforming FDIR (0.86) and EQS (0.86), while sustaining the lowest runtime of 0.84 s. As document volume scaled from 100 K to 1,200 K, document quality declined for all methods; however, DIRBFA maintained the highest quality score at every level, achieving **0.36** at 100 K documents. CPU runtime analysis across user requests (5–20) showed that DIRBFA was fastest or near-fastest in nearly all cases, reaching 0.74 s at 20 requests. In iterative testing (10–50 iterations), DIRBFA's F-measure improved from 0.46 to **0.71**—the highest convergence trajectory—while its CPU time declined steadily to 0.65 s, validating both accuracy and computational efficiency.

5. Conclusion

The DIR is the progression of acquiring the needed information, which is acquired based on the user query. Various approaches proposed to solve the DIR and the computational complexity is high when the size of the document or the collection of the document is high. The drawbacks identified in the available approach is rectified with the help of bio-inspired algorithms. Hence, BFA is incorporated with the DIR that rectifies the computational complexities. The experimental studies of BFA have outlined that the redundancy of elements and the unwanted elements are removed. The proposed algorithm attained better results with minimal execution time. The proposed Document Information Retrieval based Bacterial Foraging Algorithm (DIR-BFA) has shown promising results.

Reference

1. Manning, C. D., Schütze, H., & Raghavan, P. (2008). *Introduction to information retrieval*. Cambridge university press.
2. Harman, D. (2019). Information retrieval: the early years. *Foundations and Trends® in Information Retrieval*, *13*(5), 425-577.
3. Leuski, A. (2001, October). Evaluating document clustering for interactive information retrieval. In *Proceedings of the tenth international conference on Information and knowledge management* (pp. 33-40).
4. Bozorg-Haddad, O. (2018). *Advanced optimization by nature-inspired algorithms*. Springer Nature Singapore Pte Ltd..
5. Babashzadeh, A. , Daoud, M. ,& Huang, J. (2013). Using semantic-based association rule mining for improving clinical text retrieval. In *International conference on health information science* (pp. 186–197). Springer .Babashzadeh, A. , Daoud, M. ,& Huang, J. (2013). Using semantic-based association rule mining for improving clinical text retrieval. In *International conference on health information science* (pp. 186–197). Springer .
6. Zhong, N. , Li, Y. ,& Wu, S.-T. (2012). Effective pattern discovery for text mining. *IEEE transactions on knowledge and data engineering, 24* (1), 30–44 .
7. Menezes, G. , Almeida, J. , Belém, F. , Gonçalves, M. , Lacerda, A. , de Moura, E. , et al. (2010). Demand-driven tag recommendation. *Machine Learning and Knowl- edge Discovery in Databases* , 402–417 . Niknam, T. ,&Golestaneh, F. (2013). Enhanced
8. Lan, M. , Tan, C. L. , Su, J. ,& Lu, Y. (2009). Supervised and traditional term weighting methods for automatic text categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 31* (4), 721–735 .
9. Veloso, A . A . , Almeida, H. M. , Gonçalves, M. A. , & Meira Jr, W. (2008). Learning to rank at query-time using association rules. In *Proceedings of the 31st annual in- ternational ACM SIGIR conference on research and development in information re- trieval*(pp. 267–274). ACM .
10. Khennak, I. ,&Drias, H. (2016). A firefly algorithm-based approach for pseudo-rel- evance feedback: Application to medical database. *Journal of Medical Systems, 40* (11), 240 .
11. Lin, C.-H. , Chen, H.-Y. ,& Wu, Y.-S. (2014). Study of image retrieval and classification based on adaptive features using genetic algorithm feature selection. *Expert Sys- tems with Applications, 41* (15), 6611–6621 .
12. Niknam, T. ,&Golestaneh, F. (2013). Enhanced bee swarm optimization algorithm for dynamic economic dispatch. *IEEE Systems Journal, 7* (4), 754–762 .
13. Jung, J. J. (2012). Evolutionary approach for semantic-based query sampling in large-scale information sources. *Information Sciences, 182* (1), 30–39 .

14. Akbari, R. , Mohammadi, A. ,&Ziarati, K. (2010). A novel bee swarm optimization al- gorithm for numerical function optimization. *Communications in Nonlinear Sci- ence and Numerical Simulation, 15* (10), 3142–3155 .
15. Robbi Rahim. (2026). Decentralized Peer-to-Peer Renewable Energy Trading Using Blockchain and Smart Contracts. *National Journal of Renewable Energy Systems and Innovation* , 1-8.
16. K P Uvarajan. (2026). Impact Assessment of Conservation Tillage Systems Using Remote Sensing and Agro-Environmental Modeling. *Journal of Environmental Sustainability, Climate Resilience, and Agro-Ecosystems*, 1–8.
17. M. Kavitha. (2025). Neural Mechanisms of Attention and Inhibition in Complex Decision-Making Tasks: An fMRI-Based Analysis. *Advances in Cognitive and Neural Studies*, *1*(3), 1-7.
18. Aakansha Soy, & Mrunal Salwadkar. (2025). Systems Biology Framework for Predicting Drug Toxicity in Preclinical Studies. *Frontiers in Life Sciences Research*, 22–31.