AN EFFECTIVE DOCUMENT INFORMATION RETRIEVAL USING ENHANCED MAP REDUCE BASED CLUSTERING

Selvi P, Assistant Professor, Department of Computer Science,
KG College of Arts and Science, Coimbatore.
p.selvi@kgcas.com, selviragu98@gmail.com

Abstract:

The organizations make use of the Information retrieval techniques, in order to ease the search for information. Recognizing the documents from the collection is nothing but the Document Information retrieval, that too which are most relevant to a user query. The data mining techniques are used in the preprocessing step in the current methodology for dividing the document collection and it drew-out the most closed frequent terms on each cluster already created. But, here we have few disadvantages which explore the advances in the data mining field for rectifying the fundamental Document Information Retrieval problem. In our proposed work, data mining concept assist us in getting the useful knowledge and this knowledge was utilized by swarms for exploring the entire space of documents in an intelligent manner. Enhanced Map Reduce Algorithm was proposed in this work for rectifying the above mentioned issue and also to extract the most closed frequent terms on each cluster. Anarchies Society Optimization (ASO) was proposed finally for exploring the document clusters which has been created previously with the help of Enhanced Map Reduce Algorithm for any user's request. The proposed approach has been computed on well-known collections like CACM (Collection of ACM), TREC (Text REtrieval Conference), Webdocs, and Wikilinks, and it has been distinguished with the state-of-the-art data mining techniques.

Keywords---- Information retrieval, Data mining, Big data analysis, Swarm algorithm**,** Anarchies Society Optimization (ASO).

I. INTRODUCTION

In order to extract the useful patterns in text documents we make use of the data mining concept. Recognizing the interesting knowledge in the text documents were done in text mining and it is a demanding issue for recognizing the exact knowledge in text documents, which assist the users to find their exact requirements. Data mining techniques helps in text analysis by extracting occurring terms as descriptive phrases from document groups [1].

So, we consider data mining as a significant step in the process of knowledge discovery in databases, which means: data mining comprises entire methods of knowledge discovery process and performing modeling phase that is an application of methods and algorithm for calculation of search pattern or models.

The interesting knowledge was recognized through Text mining in text documents. Recognizing the exact knowledge in text documents assists the users to discover what they want, and it is a great dispute. Many term-based methods were supplied by Information Retrieval (IR) to rectify this challenge, Earlier, Term based methods comprises of various advantages like efficient computational performance as well as mature theories for term weighting, which have appeared over the last couple of decades from the IR and machine learning section. Polysemy and synonymy gives much issue. The former one is: a word has many meanings, whereas the latter one:

many words having the same meaning. Various semantic meaning was uncertain for responding the exact requirement. Numerous data mining methods were evolved in the last decade and in text field, mining is a tedious process and it is very ineffective through this discovered knowledge. Then due to some helpful pattern, we have much support for specificity. Data mining techniques tends to the ineffective performance and it was derived by misinterpretations of patterns [2]. An effective pattern discovery technique helps to rectify the low-frequent and misinterpretation problems for text mining. And here we make use of two processes. Further we make use of pattern deploying and pattern evolving for refining the discovered patterns in text documents.

Two main approaches were proposed in this work. The works reported in [3] use the K-means algorithm [4] to group the clusters into k disjoints clusters, where every group comprises of similar documents. While, the works reported in [5] makes use of Frequent Patterns Mining (FPM) [6] for recognizing the frequent terms in the collection. Then, the top k frequent patterns help to generate the groups of documents. These approaches decompose the initial problem into various sub-problems, where each of which could be rectified independently. But, the runtime of the DIR problem is still excessive, particularly while dealing with massive number of documents existing in the World Wide Web (WWW). Here, the proposed work enhances the pre-processing step of the current information retrieval approaches by enforcing both clustering and closed frequent itemset mining to extract knowledge from a collection of documents. Certainly, the K-means algorithm produces *k* clusters; then FPM is performed on every cluster to extract the frequent patterns among the highly correlated documents. The proposed systems bring-in the Enhanced Map Reduce Algorithm to extracts the most closed frequent terms on each cluster. Finally, ASO helps to explore efficiently the documents which were generated previously through Enhanced Map Reduce Algorithm for any user's request. The experimental analysis reveals that our approach beats the data mining based approach with huge collections.

I. RELATED WORKS

TDC algorithm was given by Yu et al. [7], which mines patterns in documents for enhancing the quality of document classification. Further it creates the topics which explain the documents through only the closed frequent item sets. And it addresses that TDC is faster when compared with FIHC for answering queries. TDC makes use of a structure, which allows to hierarchically constructing links among each itemset of a same size k using itemsets of size k-1. High precision will be given by this approach. But, the clusters generated by TDC overlap when terms recognized in documents were highly correlated.

An algorithm for text processing called ARMIR (Association Rule Mining for Information Retrieval) was given by Babashzadeh et al. [8]. A user query was modeled by this approach, as a set of ideas where relationships among concepts are determined by association rule mining.

A ranking function was designed by a ranking function for ranking the document. The approach comprises of first mining rules from a set of training documents. The resulting of a rule represents the scores of documents enclosing the terms appearing in its antecedent.

PTM (Pattern Taxonomy Mining) algorithm was designed by Zhong, Li, and Wu (2012) [10], for enhancing the comprehension of the user's request using a patterns mining algorithm. BY enforcing the closed algorithm in the training set of documents, the pattern taxonomy of terms was discovered. The noise between the user's request and the set of terms in the collection of documents will be minimized by this technique.

In [11], a new supervised term weighting approach called KNNIR (K-Nearest Neighbors for Information Retrieval) is proposed, whihc combines the support vector model representation and KNN (K-Nearest Neighbors) algorithm for evaluating the weight of each term in the given documents. The weights of the training terms were initially measured, then the KNN classifier is launched for measuring the score between each test term and the training terms.

II. PROPOSED METHODOLOGY

In fig 1, the research work of our proposed work is given. The primary target of this work is to develop the power of data mining techniques for drawing-out an appropriate knowledge, which will be utilized later by the swarms and further this approach performs two main stages. The initial stage is to divide the collection of documents into various clusters, where every cluster can be viewed as a subset of documents of the entire collection of documents. The set of terms shared by two clusters is known as separator set. Reducing the size of separator sets while arranging the same cluster documents which has been connected, i.e., documents that share the maximum number of terms, was considered as the interesting decomposition approach. A partitioning- based strategy by adapting K-means algorithm has been proposed in this work for decomposing the whole collection of documents. The result of this step is: it generates the cluster, each of which has a subset of documents highly connected. Further, we enforce the Frequent Patterns Mining (FPM) approach to every cluster of documents. Here, the closed frequent patterns were developed each cluster. Then an enhanced map reduce algorithm was proposed to discover the closed frequent patterns of every group of documents highly connected. The optimization algorithm makes use of the closed frequent patterns already extracted on each cluster to explore the solution's space. Alternatively, the swarms were guided by the closed frequent patterns discovered during the preprocessing step when exploring the solution's space. Two strategies were established to show the value of the discovered patterns for guiding the search of swarms. It is worth to address that several swarm-based approaches can be investigated here.

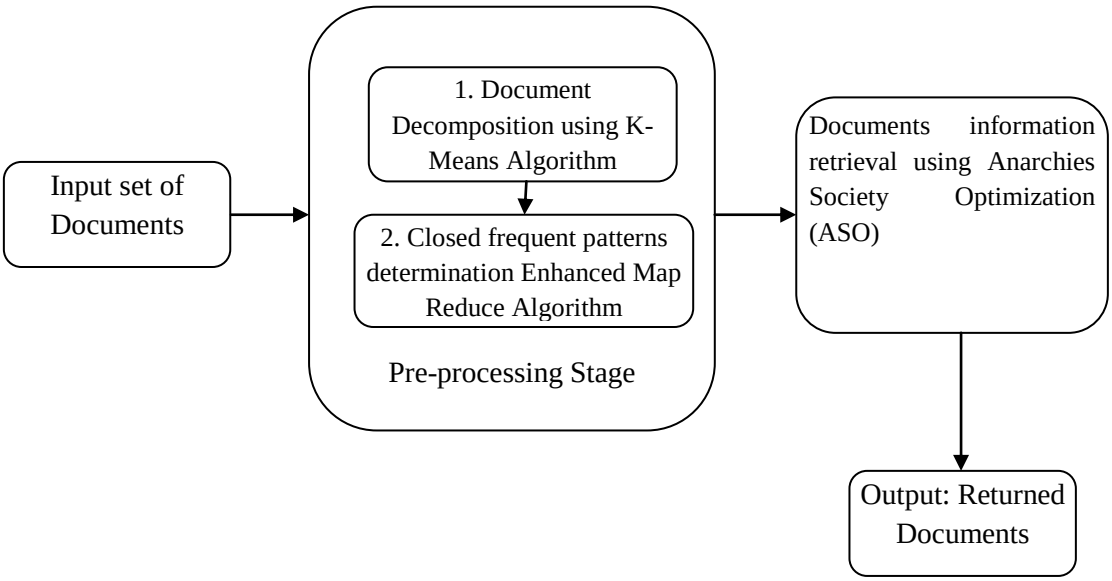

Figure 1: Overall proposed methodology diagram

A. Preprocessing Steps

This step involves two stages (document decomposition and closed frequent patterns determination) which are given as follows:

i. Documents decomposition using k means algorithm

K-means helps to decompose a given collection of documents without loss of generality at the initial stage and it is considered as the simplest unsupervised learning algorithms for the clustering problems. Further it determines a simple and effortless procedure to split a provided data set into a certain number of clusters, say k clusters, fixed a priori. The primary target of this work is to determine k centroids, one for every cluster. The clustering result depends on their location in the clusters, because the centroids should be placed in a cunning way. It is judicious to place them as far as possible from each other, in order to optimize the result. The next step is to consider every point which belongs to a provided data set and it gets connect with the nearest centroid. The first step was assumed as completed, if no point is in pending status and an early grouping is performed. Here, k new centroids were required for re-evaluating the new clusters that result from the previous step, and the process should be iterated. The latter stops when no more changes of the clusters are observed, i.e., when centroids stop moving. In this work, we used the adaptation explored in [3] for its simplicity and efficiency.

The general algorithm of k-means could be represented as follows.

$$J = \sum_{j=1}^{K} \sum_{n \in S_j} \left| X_n - \mu_j \right|^2 \quad (1)$$

Where

X_n is a vector that represents the *n* th data point

μ_j is the centroid of the data points in S_j

Partitioning (or clustering) N data points into K disjoint subsets S_j were the target of -means algorithm. Every subset S_j comprises of $N_j$data points. The target is to minimize the sum-of-squares criterion.

Initially, the data points were assigned randomly to the K clusters and the centroid was evaluated for each cluster. Further, each point was allocated to the cluster whose centroid is the closest to that point. These two steps were iterated till there is no further assignment of the data points to the clusters. In the following, we indicate the adaptation of K-means to our problem.

Documents Representation

The documents were indicated through the vector space model. Each document d is indicated by a vector $\{w_1, w_2 \ldots . w_n\}$, where w_i indicates the weight of the term t_i . The term weight value indicates the importance of this term in a document which is computed through the well- known T F −IDF (Term Frequency with Inverse Document Frequency) formula [12] as follows:

$$w_{ij} = tf_{ji} \times idf_{ji} \quad (2)$$

Where w_{ij} indicates the weight of the term *i* in the document *j* . tf_{ji} is the number of occurrences of term *i* in the document *j* . $idf_{ji} = \log_2(\frac{m}{df_{ji}})$ such df_{ji} represents the term frequency in the collections of *m* documents.

Similarity Computation

The similarity computes among two documents d_i and d_j is computed with the help of the cosine correlation measure Tata and Patel (2007) given by:

$$\cos(d_i, d_j) = \frac{d_i^t d_j}{|d_i||d_j|} \quad (3)$$

Where $d_i^T d_j$ indicates the dot-product of the two document vectors d_i and d_j . $|d_i|$ indicates the length of the vector d i , i.e., the number of terms having weights non null in the document d_i .

Centroids Updating

The centroid updating was measured as:

$$g_i = \frac{1}{|c_i|} \sum_{j=1}^{|c_i|} d_j \qquad (4)$$

Where: g_i is the new center of the cluster C_i.

ii. Closed frequent patterns mining using enhanced Map reduce algorithm

Recognizing the interesting patterns from the given input data was the main target of Frequent Patterns Mining (FPM), basically it creates a high number of patterns, particularly when the minimum support is low. Some other approaches permit this to minimize the dimensional- ity of the resulted frequent patterns significantly. Amid them, we address the closed frequent patterns mining which extracts only the closed frequent patterns. If no superset of this pattern with the same support exists, we name it as frequent pattern. In [13], the author explained the definitions of the closed frequent pattern problem. Here, various algorithms for solving Closed FPM problem have been explored, which develops various optimizations to save both space and time in enumerating the closed frequent patterns. The following algorithm recognizes the closed frequent patterns from each cluster of documents through the enhanced map reduce algorithm.

Enhanced Map Reduce Algorithm

The K-means clustering algorithm is process by using MapReduce can be divided into the following phases:

1. Initial

(i) The given input data set can be split into sub datasets. The sub datasets are formed into <Key, Value> lists.

And these <Key, Value> lists input into map function.

(ii) Select k points randomly from the datasets as initial clustering centroids.

2. Mapper

a) Update the cluster centroids. Calculate the distance between the each point in given datasets and k centroids.

b) Arrange each data to the nearest cluster until all the data have been processed.

c) Output pair $< c_i, z_j >$. And c_i is the center of the cluster z_j.

3. Reducer

(i) Read $< c_i, z_j >$from Map stage. Collect all the data records. And then output of k clusters and the data points.

(ii) Calculate the average of each cluster which is selected as the new cluster center.

Initializing the position of the clusters

Here we assume the Forgy method to set the positions of the k clusters to k observations selected randomly from the dataset.

Since the algorithm stops in a local minimum, the initial position of the clusters is very important.

B. Anarchies Society Optimization (ASO) for document information retrieval

Let $D = \{d_1, d_2 \ldots d_m\}$ be the set of m documents and let $T = \{t_1, t_2 \ldots t_n\}$ be the set of n terms. Every document is composed of the subset of terms included in T . The user's request Req is indicated by the set of terms. The Document Information Retrieval (DIR) is the process of recognizing appropriate documents, based on the user's request Req, from a collection of documents D. Here, an ASO algorithm is proposed for recovering the information from the documents in an effective way by reducing the runtime etc.

Swarm intelligence (SI) unease the collective, rising behaviour of multiple, interacting agents who

follow some simple rules. Whereas every agent may be assumed as unintelligent, the entire system of multiple agents may show some self-organization behaviour and therefore it can behave like some sort of collective intelligence. With the help of drawing inspiration from swarm-intelligence systems in nature, various algorithms have been established. Natural evolution based heuristic optimization methods such as genetic algorithms, evolutionary programming, differential evolutionary and swarm-intelligence based random optimization algorithms like Particle Swarm Optimization (PSO) and Ant Colony Optimization (ACO) helps to rectify the tedious combinatorial optimization problems more than three decades. On simulating natural, social insect colonies and animal group behavior, these methods were utilized for deducing the fundamentals of self-organization and cooperation. The structure of a nature-inspired swarm intelligence method strongly works according to the personal and social characteristics of its population's members. Hence, selecting an appropriate underlying society is significant for establishing such algorithms. Anarchic Society Optimization (ASO) established by Ahmadi-Javid[8] was a initial brought-in human-inspired swarm intelligence optimization method. This novel random method works according to an abnormal human society rather than a swarm of birds or a colony of ants, which are the basis of PSO and ACO, correspondingly. It is addressed that the humanity has very special and unique characteristics when compared with the social insect colonies and animal groups. ASO is a modern optimization method inspired by a human society whose entities behave anarchically to improve their situations. In ASO algorithm the entities are indecisive and their irregularity maximizes their situation worsens. They also behave illogically and bravely, navigating to the inferior positions they have visited. With the help of these chaotic members, ASO is able to search the solution space perfectly and keep away from falling into local optimum solutions.

Consider that S be a solution space and $f(.)$ be a cost function requires to be minimized over S. Assume a society of N members searching within an unknown land, i.e. the solution space, for the best place to live, i.e. the global minimizer of f over S. The primary feature of the society is that its members are brave and behave anarchically at the time of their search process. Assume $X_i(k)$ is the position of member i in iteration k of the exploring procedure. All of the members are aware of the best global position $(G(k))$ which is known as G-best visited by the whole society in the first k iterations. They also realize member i_k^* who occupies the best position in the society in iteration k. The best personal position (P-best) previously visited by member i in iteration k was indicated by $P_i(k)$.

Every member has a planning process to decide how it will navigate and change his/her position in the next time. To this end, every member gives three movement policies and then it connects them to recognize out his/her position in the next iteration. These movement policies were explained in three cases as follows:

a) Movement policy choice based on the current position:

The initial movement policy in time k is represented by $MP_i^{current}(k)$ and is selected based of the current position. Specifically, the movement policy $MP_i^{current}(k)$is a neighboring method. The fickleness index $FI_i(k)$ is assumed for member i in time k. This index computes the member i's dissatisfaction for his/her current situation, distinguished to other member's situations. When

the objective function f is positive on S, $FI_i(k)$for some nonnegative number $\alpha_i$in [0,1] is given as follows

$$FI_i(k) = 1 - \alpha_i \frac{f(X_{i_k}^*(k))}{f(X_i(k))} - (1-\alpha_i)\frac{f(P_i(k)?}{f(x_i(k))} \quad (5)$$

$$FI_i(k) = 1 - \alpha_i \frac{f(G(k))}{f(X_i(k))} - (1-\alpha_i)\frac{f(P_i(k)?}{f(x_i(k))} \quad (6)$$

which are the numbers in the interval [0,1].

b) Movement policy choice based on other members' positions:

The second movement policy in time k is represented by $MP_i^{society}(k)$and is chosen according to the positions of the other members. It is logical and regular that each member would create his/her movement policy $MP_i^{society}(k)$based on G-best (or position of member i_k^*); however, because members were irregular and adventurous, they may choose any one of the other members' positions (or a number of them) to create a movement policy. Therefore, the external irregularity index E$I_i(k)$is determined for member i in time k which can be used in two cases:

- In the first case, $EI_i(k)$ it is assumed as the probability that member i will behave unevenly and create his/her movement policy according to another randomly selected member's position, which doesn't correspond to G-best.

● In the second case, $EI_i(k)$ it was distinguished with a threshold. Member i will behave irregularly, if it is greater than the threshold,. The number $EI_i(k)$can be determined based on member i's situation relative to G-best (or the i_k^* th member's position) for some positive number θ_i as follows:

$$EI_i(k) = 1 - e^{-\theta_i[f(x_i(k))-f(G(k))]} \quad (7)$$

c) Movement policy choice based on past positions:

The third movement policy in time k is indicated by $MP_i^{past}(k)$and is chosen according to the past positions that were visited by each member. It would be more common for every member to create the movement policy $MP_i^{past}(k)$based on P-best, but because the members are lawless, they may choose any past position (or a number of them) to create a movement policy. Therefore, internal irregularity index $II_i(k)$ was brought-in for member i in time k, which helps the scenarios indicated for $EI_i(k)$in the previous case

2.3. Combination Rule After selecting movement policies $MP^{current(k)}$, $MP^{society(k)}$, $MP^{past(k)}$, each member must combine these policies to navigate toward a new position, so he/she requires a combination rule. The simplest method to this is to choose the movement policy that gives the best new position, which is termed as elitism combination rule. An option is that the policies were enforced successively on the current position; this may be termed as the sequential combination rule. Other types of combination rules can be determined based on the problem definition. Note that members may use different combination rules

Pseudo code of ASO Algorithm to retrieve the document information

Input: The set of Clusters of Documents $C = \{C_1, C_2 \ldots C_k\}$

The set of Closed Frequent Patterns $F = \{F_1, F_2 \ldots . F_k\}$

The User's Request Req .

Begin

for each cluster C_i do

Initialize members of society randomly

Planning for movement based on current position

Compute fickleness index

Compute external irregularity index

Compute internal irregularity index

Select the movement policy based on the three $MP^{current(k)}$, $MP^{society(k)}$, $MP^{past(k)}$

{

for

$MP^{current(k)}$, based on fickleness index

$MP^{society(k)}$, based on external irregularity index

$MP^{past(k)}$ based on internal irregularity index

End for

}

Update position by combining all the movement policies

End if it meets the stopping condition

Or else

Update the each member position based on the cluster value

End for

The matching procedure was initially enforced to each cluster of documents and the user's request, based on this algorithm. This procedure returns the number of common terms among the closed frequent patterns of every cluster and the user's request. The likelihood of choosing the documents of every cluster is then allocated with the help of the matching function and the number of all terms n. At last the optimization algorithm, the initial member of society generates the initial solution; which defines the navigation according to the existing position by assuming the probabilities of entire documents already. At the end of each pass of the algorithm, the best solution will be updated position for the next iteration. This process should be iterated till the maximum number of iterations is reached.

III. EXPERIMENTAL RESULTS

In order to explain the performance of the proposed approach in the DIR problem, various experiments were done. Basically we explain the collections of documents which help in the experiments, followed by a discussion of how the parameters in the suggested approach have been fixed. Then, we distinguish the pro- posed approach with existing bio-inspired approaches for dealing with DIR problem with respect to runtime performance and the quality of returned documents. Regarding the evaluation measure, we have used measure it is according to (Recall and Precision) and it is the well-known measure for the DIR problem:

Recall is the ratio of the number of appropriate documents retrieved to the total number of all relevant documents.

$$Recall = \frac{|RDR|}{|ARD|} \quad (8)$$

Where: RDR: The set of the Relevant Documents Retrieved,

ARD: The set of All Relevant Documents.

Precision is the ratio of the number of relevant documents retrieved to the total number of returned documents.

$$Precision = \frac{|RDR|}{|RD|} \quad (9)$$

Where: RD: The set of all returned documents.

F-measure This measure allows to combine precision and recall measures, which is defined as follows:

$$F - measure = \frac{2 \times Recall \times precision}{Recall + Precision} \quad (10)$$

4.1. Collection of documents

Based on size, the collection in our evaluation varies, i.e., we consider medium, large and big collections. The first collection we utilize her is CACM and it is a collection of abstracts of articles available in CACM journal between the years 1958 and 1979. And further it comprises of 3204 documents and 6468 terms and it is assumed as a medium collection. The second set of data instances that we assume is a large collection retrieved from the renowned TREC repositories. It was basically generated in 1992 by the U.S. National Institute of Standards TREC and Technology (NIST). Within this framework, there have been many tracks over a range of different topics including Ad Hoc, Medical, Weblogs, and Others.

Table 1 Quality of returned documents of Proposed ASO using CACM collection with different number of clusters.

Number of Clusters	BSOGDM	Proposed ASO
5	0.42	0.51
10	0.54	0.62
20	0.74	0.75
30	0.74	0.78
40	0.74	0.82
50	0.74	0.82

Table 2 Runtime (sec) of Proposed ASO using CACM collection with different number of clusters.

Number of Clusters	BSOGDM	Proposed ASO
5	0.16	0.12
10	0.25	0.21
20	0.31	0.25
30	0.51	0.45
40	0.91	0.83
50	1.11	1.01

Tables 1 and 2 reveals that both the quality of returned documents (F-measure) and the runtime (in Sec) of the proposed ASO and the existing BSOGDM using CACM collection with various clusters and various minimum support. The quality of returned documents enhances until it attains the iteration number 20, by increasing the number of clusters from 5 to 50, where it stabilizes at 0.82 with slight difference in runtime. As a result, the number of clusters is set to 20 for the rest of the experiments. Again, by maximizing the minimum support from 10% to 100%, the quality of returned documents stabilizes at 0.82 until the iteration number 50%. Further, the quality starts decreasing from 60% to 100%. Yet, the runtime is minimized while the minimum support is decreased. Therefore, the minimum support is set to 50% for the remaining of experiments.

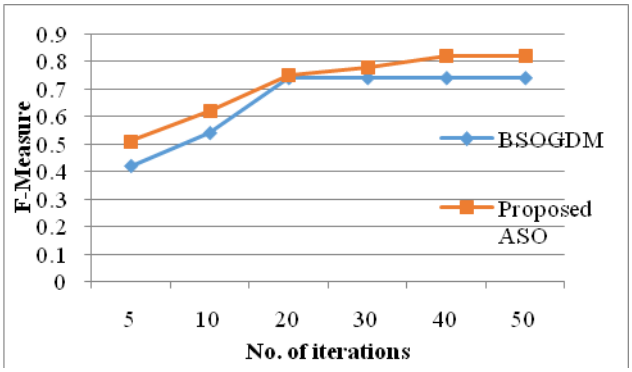

Figure 2: F-Measure Comparison

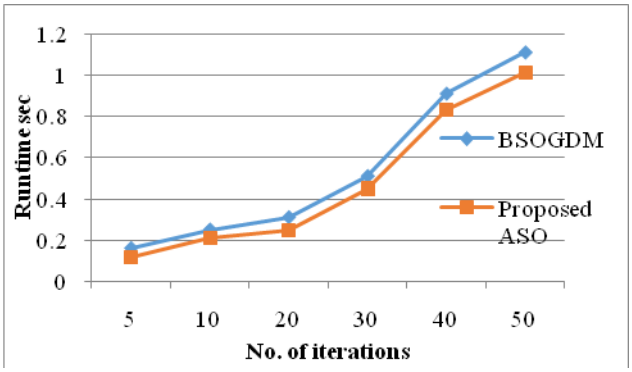

Figure 3: Runtime Comparison

Figure 2 and 3 provides the quality of returned documents (F-measure) and the runtime (in Sec) of the proposed ASO and the existing BSOGDM.

IV. CONCLUSION

For solving the DIR problem, here the author proposed the new swarm intelligence approach and this swarm explores the space of documents through knowledge discovery by two data mining techniques. Initial one decomposes the entire collection of documents into similar and disjoint clusters with the help of the K-means algorithm. And further it basic aspect is utilization of a distance measure between documents, and a technique to define the centroid for the set of documents. The other one extracts the frequent terms from each cluster of documents with the help of the enhanced map reduce algorithm. Utilizing the minimum supports which controls the set of frequent terms extracted, is main specificity of this approach. So, the collection of documents gets classified into various sub-collections, with the help of these two techniques, each of which is featured by the set of closed frequent terms among the documents belonging to it. Then for exploring each cluster deeply for any user's request, the swarms utilize this knowledge. The experimental analysis says that our approach beats the other algorithms with respect to document's quality and has a very competitive run time.

V. REFERENCES

1. Stafylopatis A, Likas A. Pictorial information retrieval using the random neural network. IEEE Transactions on Software Engineering. 1992; 18(7):590–600.
2. Wives LK, Loh S. Hyperdictionary: A knowledge discovery tool to help information retrieval. 1998 Proceedings String Processing and Information Retrieval: A South American Symposium; Santa Cruz de La Sierra. 1998. p. 103–9.
3. Mahdavi, M. , & Abolhassani, H. (2009). Harmony k-means algorithm for document clustering. Data Mining and Knowledge Discovery, 18 (3), 370–391 .
4. MacQueen, J. (1965). On convergence of k-means and partitions with minimum av- erage variance. In Annals of mathematical statistics: 36 (p. 1084). Inst Mathemat- ical Statistics IMS Business Office-Suite 7, 3401 Investment Blvd, Hayward, CA 94545 .
5. 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 .
6. Zaki, M. J. , & Hsiao, C.-J. (2002). Charm: An efficient algorithm for closed itemset mining. In Proceedings of the 2002 SIAM international

conference on data mining (pp. 457–473). SIAM .

7. Yu, Searsmith, D., Li, X., & Han, J. (2004,Scalable construction of topic directory with nonparametric closed termset mining. In Data Mining, 2004. ICDM'04. Fourth IEEE International Conference on (pp. 563-
8. Babashzadeh, A., Daoud, M., & Huang, J. (2013). Using semantic-based association rule mining for improving clinical text retrieval. In Health Information Science (pp. 186-197). Springer Berlin Heidelberg
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. 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 .
11. 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 .
12. Blei, D. M. , Ng, A. Y. , & Jordan, M. I. (2003). Latent dirichlet allocation. Journal of Machine Learning Research, 3 (Jan), 993–1022 .
13. Youcef Djenouri a , ∗, Asma Belhadi b , Riadh Belkebir, " Bees swarm optimization guided by data mining techniques for document information retrieval", Expert Systems With Applications 94 (2018) 126–136.