BEES ALGORITHM-BASED DOCUMENT CLUSTERING

Nihal M. AbdelHamid[1], M. B. Abdel Halim[1], M. Waleed Fakhr[2]
[1]Arab Academy for Science, Technology and Maritime Transport, College of Computing and Information Technology, Cairo, Egypt
[2]University of Bahrain, Dept. of Electrical Engineering, Manamah, Bahrain
[1]nihalali@eg.ibm.com, mbakr@ieee.org
[2]waleedf@aast.edu

Abstract

Document clustering is one of data mining fields which mainly aims to automatically group the relevant documents into clusters. This paper introduces the application of the Bees Algorithm in optimizing the document clustering problem. The Bees Algorithm simulates the foraging behavior of honey bees swarms in collecting nectar from flower patches by performing global and local search simultaneously; this improves avoidance of local minima convergence. Genetic Algorithm and the K-means algorithm are used for comparisons as they are among the most commonly used techniques to solve the problem.
Many experiments were performed on a corpus of 818 documents from 4 different fields and the results have shown that the Bees algorithm outperforms the Genetic Algorithm and the K-means by 15% and 50% respectively, in terms of solutions fitness, with an acceptable increase in the processing time.

Key Words - Document Clustering, Bees Algorithm, Genetic Algorithm, K-means, Evolutionary Algorithms

1 INTRODUCTION

A clustering algorithm attempts to find natural grouping of a given data points based on the similarity between these points. Moreover, the clustering algorithm finds the centroid of a group of data sets. To determine cluster membership, most algorithms evaluate the distance between a point and the cluster centroids. The output from a clustering algorithm is basically a statistical description of the cluster centroids with the number of components in each cluster. Figure 1 simply illustrates the clustering process.

RAW DATA
CLUSTERING ALGORITHM
CLUSTERS OF DATA

Figure1. Clustering Process

Several Clustering algorithms have been developed [1] yet most of them could not fulfill the requirements of document clustering problem which are:

a. **High dimensionality**: The number of relevant terms in a document set is typically in the order of thousands, if not tens of thousands. Each of these terms constitutes a dimension in a document vector. Natural clusters usually do not exist in the full dimensional space, but in the subspace formed by a set of correlated dimensions. Locating clusters in subspaces can be challenging.

b. **Scalability:** Real world data sets may contain hundreds of thousands of documents. Many clustering algorithms work fine on small data sets, but fail to handle large data sets efficiently.

c. **Accuracy:** A good clustering solution should have high intra-cluster similarity and low Inter-cluster similarity, (i.e., documents within the same cluster should be similar but are dissimilar to documents in other clusters).

The following section introduces several techniques used to solve optimization problems; That are the problems with no defined best solution, the document clustering problem is a very good example for the optimization problems. The Bees Algorithm (BA) used in this paper belongs to the evolutionary algorithms family where a problem is optimized by iteratively enhancing candidate solutions with respect to an evaluation metric. BA is a population-based algorithm that performs a combination of local and global search in a way that simulates the foraging behavior of the honey bees swarms.

To measure the efficiency of the algorithm when applied to document clustering problem [2], comparisons are made between BA and the Genetic Algorithm (GA) as well as the K-means algorithm.

The paper is organized as follows; a brief background discussion on the K-means and natural-inspired optimization techniques are shown in sections 2 and 3 respectively. Section 4 shows how the document similarity is calculated before applying a clustering algorithm to the dataset. Section 5 explains the Bees algorithm and document clustering modeling to the bees algorithm. The Genetic Algorithm is discussed in section 6, while in section 7 the experimental results are illustrated. Finally section 8 concludes the paper and discusses some future plans.

2 Background

Several Clustering algorithms have been developed [1-2]; Traditional document clustering is usually treated as an unsupervised learning problem (*i.e. only unlabeled documents are taken as input*). Usually, document clustering techniques fall into three categories: Graph-based, Hierarchical and Partitioning clustering.

1. **Graph-based methods** model the input documents as vertices of a weighted graph; the edge weight is given by the similarity between two corresponding documents. Accordingly, document clustering problem is turned to graph partitioning based on a certain criterion.
2. **Hierarchical clustering,** this technique successively builds a tree-like clusters structure which can be constructed in either divisive (top-down) or agglomerative (bottom-up) manner. The main problem with hierarchical clustering is that, it is non-scalable which makes it not suitable for real-time applications and large corpora.
3. **Partitioning clustering** is to directly partition a dataset into K groups such that documents in a group are more similar to each other than any document from another group. The disadvantage of this technique is it can converge to a sub-optimal solution. K-means is a typical example to this kind [1].

K-means [1] is one of the most common partitioning clustering algorithms because it is simple to implement and has a very convenient computational efficiency –as it linearly grows with the number of data points- this linear growth makes K-means very convenient to large data sets. The main drawback of K-means is that it can easily converge to suboptimal local minima.

In **K-means** algorithm, a cluster is represented by the mean value of data points within a cluster (i.e. Centroid) and the clustering is done by minimizing the sum of Euclidean distances between data points and the corresponding cluster centroid. This minimization criterion is shown in equation (1).

$$\arg_S \min \sum_{i=1}^{k} \sum_{x_j \epsilon S_i} ||x_j - \mu_i||^2 \tag{1}$$

Where X is a set of documents (x_1, x_2, x_3.... x_n) , k is the number of clusters the documents will be partitioned to, and the S is a cluster of documents S=(s_1, s_2, s_3,... s_k), μ is the center of the i^{th} cluster, and the algorithm is working in a manner that minimizes sum of squares within one cluster.

The steps of the K-means algorithm are as follows:

1. Initialize the K cluster centers randomly.
2. Decide the class memberships of the *N* objects by assigning them to the nearest cluster centroid.
3. Re-estimate the K cluster centers, by assuming the memberships found above are correct.
4. If none of the *N* objects changed membership in the last iteration, exit. Otherwise go to step 3.

The next section introduces a brief description of Swarm-Based Optimization techniques, which have different and efficient concept in solving optimization problems.

3. NATURAL-INSPIRED OPTIMIZATION TECHNIQUES

3.1 Evolutionary Algorithms

Evolutionary Algorithms (EA) are very commonly used in solving optimization problems. It emulates the natural behavior of population recombination and generating new generations (*i.e. survival for the fittest*) by means of two main operators:

- **Recombination**: Applied on two or more selected candidates that results one or more new solutions
- **Mutation:** Applied to a candidate solution to result a new candidate.

The Genetic Algorithm [3] is one of the famous evolutionary algorithms that simulates the natural chromosomes selection and recombination; GA can also be hybridized [4, 5] with other algorithms in order to enhance its performance. GA is discussed in more details in section 6.

3.2 Swarm-based Optimization Algorithms

Swarm-base Optimization Algorithms (SOAs) [6] belong to the natural-inspired optimization techniques. It simulates the nature behavior in order to find an optimal solution within a certain search space. The major issue that distinct SOAs over other local search algorithms such as hill climbing is that an iteration uses a population of solutions instead of one single solution; consequently, the output of one iteration is a population of solutions.
Particle swarm optimization (PSO) [4, 5] Ant Colony Optimization (ACO) [7] and the Bees Algorithm (BA) [8] are examples for SOAs.

Particle Swarm Optimization (PSO) [4] is an optimization procedure that was first developed by (*Kennedy, Eberhart*) simulates the behavior of groups of populations (e.g. a flock of birds) where individual solutions in a population are viewed as “particles” that evolve or change their positions over time based on a certain criterion. Each particle modifies its position (*i.e. improves the solution*) in search space according to that best position visited by itself and its neighbours.
The main problem with PSO is that it is usually converged quickly to a local optimal solution of the optimization problem being solved.

Ant Colony Optimization (ACO) [7] is another optimization technique that simulates the natural behavior of ants in finding the shortest path between their nest and food by means of a chemical substance called *pheromone*. As they move, ants deposit pheromone on the ground as a guide to the path they are following and depending on the quantity of the pheromone there is a probability that a stray ant could follow this path.
Experiments have shown that by applying the ACO on document clustering [9] it has given results near the results of PSO.
Researchers have also hybridized ACO with other techniques such as GA [10] and Tabu search [11] to improve its performance.

The **Bees Algorithm** (BA) [8] emulates the behaviour of honeybees in finding nectar in flower patches. The advantage of BA is that, it performs local and global search simultaneously which improves the fitness of the reached solutions and the avoidance of local minima trapping. The Bees algorithm is discussed in details in section 5.

4 SIMILARITY MEASUREMENT

Before applying a clustering algorithm to a corpus, prior steps are needed. This section shows how documents are represented in order to let algorithm works on the corpus and how the similarity between documents is measured.

4.1 Document Representation

In most document clustering techniques, the text documents to be clustered are represented using the Vector Space Model (VSM) [1], that is, a document is represented as a vector of *n* weighted terms called “features vector" , d={ W1, W2, W3….. , Wn}. These weights reflecting the frequency of the terms in the document are multiplied by the inverse of their frequency in the entire collection (TF x IDF). TFIDF [12,13] weighting function is defined by equations 2-3

$$tfidf(t_k,d_j) = tf(t_k,d_j)\cdot \log\frac{|T_r|}{\#T_r(t_k)} \qquad (2)$$

Where tf(t_k, d_j) denotes the frequency of the existence of a term t_k occurs in a document d_j *and* #*Tr* (t_k) is the number of documents in *Tr* in which a term t_k occurs at least once.

In order to make all weights fall within the interval [0,1], normalization needs to be applied. The cosine normalization equation is shown in equation 4.

$$\omega_{kj} = \frac{tfidf(t_k,d_j)}{\sqrt{\sum_{s=1}^{|T|}\left(tfidf(t_k,d_j)\right)^2}} \qquad (3)$$

Where *T* is the set of all terms that occur at least once in *Tr* .

4.2 Cosine Similarity Metric

To measure the similarity between two documents, a very common method is used; the cosine distance [1] given by equation 4.

$$Cos(D_j,D_k) = \frac{\sum_{i=1}^{n}\omega_{ij}\times\omega_{ik}}{\sqrt{\sum_{i=1}^{n}\omega_{ij}^2}\sqrt{\sum_{i=1}^{n}\omega_{ik}^2}} \qquad (4)$$

Where ω_{ij} is the weight of term i in document j.

5 BEES ALGORITHM (BA)

In nature, the honey bees aim to find the flowers in a patch with more quantities of nectar in order to produce more honey with less effort of search. Therefore the flowers with more nectar need to be visited by more bees.
The bees start searching by sending scout bees for random search in the patch; when these bees are back to the hive they evaluate the visited flowers based on a specific criterion. To communicate with each other and exchange information, bees perform a "waggle dance ". That is, they spread the following information during this dance:

1. The direction of the flowers patches (*the angle between the sun and the batch*).
2. The distance between the batch and the hive (*the duration of the dance*).
3. The quality of the flowers (*the frequency of dancing*).

According to this information honey bees can send bees to collect the nectar precisely by sending follower bees after the dancer bees.
The Bees Algorithm [8] is an optimization algorithm inspired by the natural behaviour of honey bees to find an optimal solution, the bees algorithm avoids getting trapped in a local minima because of its capability of performing local and global search.

5.1 Modeling problem to Bees algorithm

This section explains how the document clustering problem was modeled within the BA algorithm.
Document clustering is to find a set of clusters centroids such that the distance between a centroid and its entire cluster members is minimized and the distance between centroids within a set is maximized.
Table 1 shows how the problem is modeled to bees algorithm expressions.

Table 1. Modeling to bees algorithm

Document clustering Expression	Modeling	Notes
Solution	Set of clusters centroids	i.e. flower
Population	Set of solutions	i.e. flower patch

Fitness Function	Cosine Distance	See Equation 4
Stopping criterion	Max number of iterations	

As an evolutionary algorithm, the BA starts by creating an initial population of solutions and then evaluating them based on an fitness function; based on this evaluation, the best solutions are selected for neighbourhood search taking into consideration, assigning more bees to the solutions of the highest fitness's then select the fittest solution (i.e. bee) from each patch to construct the new population. The Algorithm works according to the following steps:

```
1. Generate initial population of solutions randomly (n).
2. Evaluate each solution using the cosine distance (equation 4) as a fitness function.
3. Repeat the following steps until stopping criterion is met
   {
      a. Select the best solutions (m) for neighborhood search (i.e. the ones with smallest distances)
      b. Assign more bees to the ones with highest fitness's (e) out of best solutions (m).
      c. Select the fittest bee from each patch.
      d. Assign the remaining bees for random search and evaluate their fitness.
   }
```

The algorithm requires the set of parameters shown in table 2 to be determined.

Table 2. Bees Algorithm Parameters

Parameter Name	Parameter Description
n	The number of the initial population
m	The number of sites selected for neighborhood search
e	The number of top fitness's sites (*elite*) among m
Nep	The number of bees recruited for e sites
Nsp	The number of bees recruited for m-e sites
Maxi	The max number of iterations (i.e. the stopping criterion)

6 GENETIC ALGORITHM (GA)

The Genetic Algorithm [3] is a very popular evolutionary algorithm that was first presented by John Holland in 1975. It is inspired by the biological evolution process. The GA considers the main operators of the evolutionary algorithms:

1. **The Variation operator:** the recombination (*i.e. crossover*) and mutation.
2. **The Selection operator:** this selection is based on a stochastic fitness function

The **selection** operator is based on a fitness function; according to this operator, the GA decides which candidates would be used for mating (i.e. recombination or crossover). The selection operator is also used in deciding which candidates would be included in the next populations and which ones would be eliminated. The crossover is applied with probability P_c in order to generate the new population.

A mutation function is applied on populations with a probability $P_{m.}$ Usually the value of P_m is low, about 1%, which improves the searching process with some kind of randomization.

The following steps show how the GA works:

```
1. Generate initial population;
2. Evaluate population;
3. Repeat until stopping criterion is met
   {
            a. Select parents for reproduction;
            b. Perform crossover and mutation;
            c. Evaluate population;
   }
```

Table 3 shows how the document clustering problem was modeled to the Genetic Algorithm

Table 3. Modeling to GA

GA expression	Modeling	Notes
Chromosome	Solution of K clusters	
Gene	A centroid of a cluster (i.e. a document)	
Crossover	Done by recombining the centroids of 2 solutions.	such that it doesn't produce an already existing solution
Mutation	Done by randomly replacing a centroid in a solution by a random (*i.e. Gaussian function*) centroid from the search space.	such that it doesn't produce an already existing solution
Evaluation function	Cosine Distance	See equation 4
Stopping criterion	Max number of iterations	

7 EXPERIMENTS AND RESULTS

The proposed Bees algorithm using the documents collections shown in table 4 and the initial values of parameters of the three algorithms are shown in table 5.
The parameter "Number of Clusters" K shown in table 5 was determined based on the number of categories the corpus actually has (i.e. 4); while the *maxi &* population size parameters were determined based on the experiments shown below.
To obtain statistically significant results from both BA and GA, each experiment has run for 10 times and the averages were calculated and shown in the next subsections.

Table 4: Testing corpus

Corpus Contents	Source	Number of documents	Size in KB	Number of terms
Politics	1. www.cnn.com 2. www.bbc.com 3. news.google.com	305	315	6081
Psychology	www.2knowmyself.com	218	205	3924
Sports	1. www.cnn.com 2. www.bbc.com 3. news.google.com	200	186	3583
Economy	1. www.cnn.com 2. www.bbc.com 3. news.google.com	95	81	1549

First, in order to determine the parameters' values of the BA, several experiments have been performed as illustrated in the following section.

Table 5: Initial values of the algorithms' parameters

Algorithm	Parameter	Value
Bees , GA and K-means	Number of Clusters K	4
	Max number of iterations (*maxi*)	80
Bees and GA	Population size *n*	100
Bees	m[15]	6
	e[15]	2
	Nep[15]	3
	Nsp[15]	2
GA	P_c [4]	0.9
	P_m [4]	0.01

7.1 Parameters Tuning Experiments

The first experiment was applied on the *m* parameter where several values were tested while keeping the values of the rest of the parameters the same as shown in table 5. The produced results shown in figure 2 shows that the resulting fitness has increased while increasing the value of the *m* until saturated when *m* is equal to 8 (i.e. 8% of the population size *n*) no change in the solution fitness was reached.

The second experiment was applied on the e parameter while retaining the values of the rest of parameters as stated in table 5 and m was set to 8; the chart in figure 3 demonstrates setting the e to different values ranging from 2 to 6 ,as setting it to 8 will not be useful, because in this e = m. As illustrated, setting the e to 2 (i.e. 25% of m) has gave the highest fitness, and when increased the e to 50% the fitness has started decreasing.

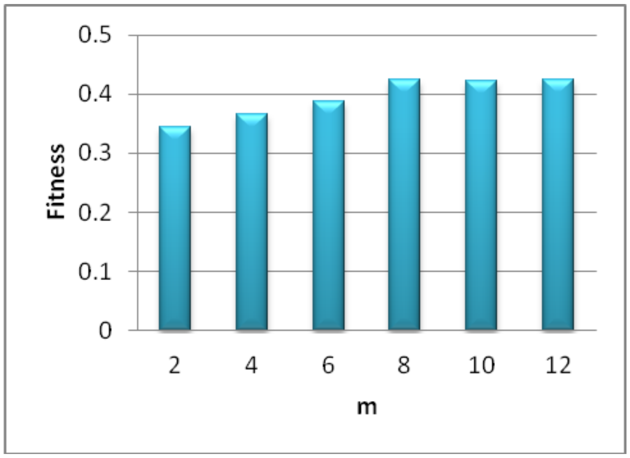

Figure 2.The m *Parameter Experiment*

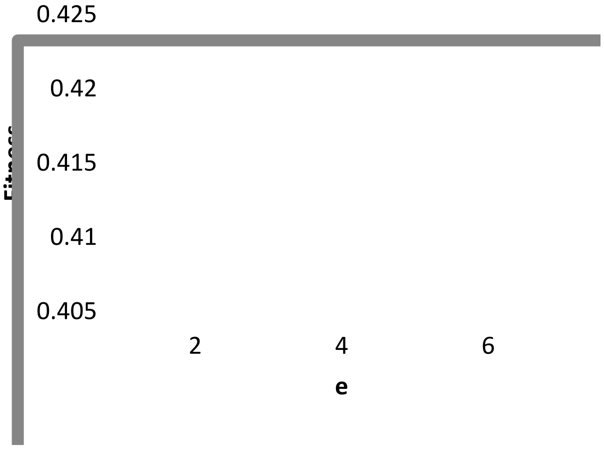

Figure 3.The *e* Parameter Experiment

The third experiment was applied on the *Nep*, as it may be noticed in figure 4, the value of 5 (i.e. 5% of the population size), has gave the highest fitness and then the fitness started to decrease on the value of 7 bees. The next experiment was applied on the *Nsp* parameter, as illustrated in figure 5 the value of 2, which represents 40% of the Nep parameter and 2% of the population size, has produced the highest fitness value and then the fitness slightly decreased and after this started to stabilize.

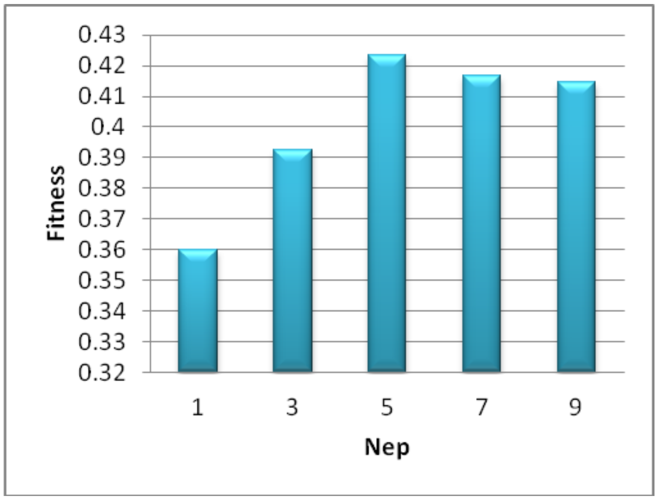

Figure 4.The Nep *Parameter Experiment*

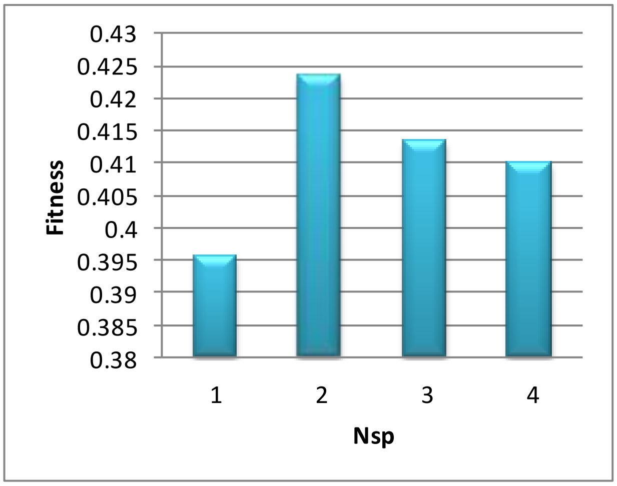

Figure 5.The Nep *Parameter Experiment*

The final values of the parameters -*after applying the experiments*- are shown in table 6.

Table 6: Algorithms Parameters and values

Algorithm	Parameter	Value
Bees , GA and K-means	Number of Clusters K	4
	Max number of iterations (*maxi*)	100
Bees and GA	Population size *n*	100
Bees	m	8
	e	2
	Nep	5
	Nsp	2
GA	P_c	0.9
	P_m	0.01

7.2 **Experiment applied on BA, GA and K-means algorithm**

This section contains the results of three experiments; two of them have been applied on the common parameters between the algorithms which are:

The maximum number of iterations (maxi) [2]: The common parameter between The Bees Algorithm, the Genetic Algorithm and K-means

Population (size) [2]: The common parameter between the Bess Algorithm and the Genetic Algorithm.

The third experiment was applied to determine the average **computation time**.

Figure 6 shows the results of the first experiment, applied on the *maxi* parameter while keeping the rest of parameters constants according to the values shown in table 6, the chart shows that the BA has reached a solution with high fitness (0.42) using less number of iterations than the GA and K-means; which shows that the BA outperforms the GA by about 15% and outperforms the K-means by about 50%. When increasing the number of iterations to more than 100 GA slightly increasing its fitness. However, the entire graph shows that the BA gives solutions with higher fitness. The chart also shows that until 300 iterations, no intersection point has been met between the two algorithms (*i.e. the GA has not reached the max fitness that the Bees algorithm has already reached yet*). The chart also shows that the K-means algorithm has the lowest fitness solutions.

The second experiment shown in **figure 7** was executed by running both of BA & GA algorithms (*because the K-means does not have many evolving solutions; hence this experiment was not applicable on it*) with different population sizes varying from 40 to 140 while keeping the rest of parameters constants as per the values in table 6; The chart in figure 7 shows that both algorithms' fitness increases when increasing population size, however the BA gives higher fitness than the GA.

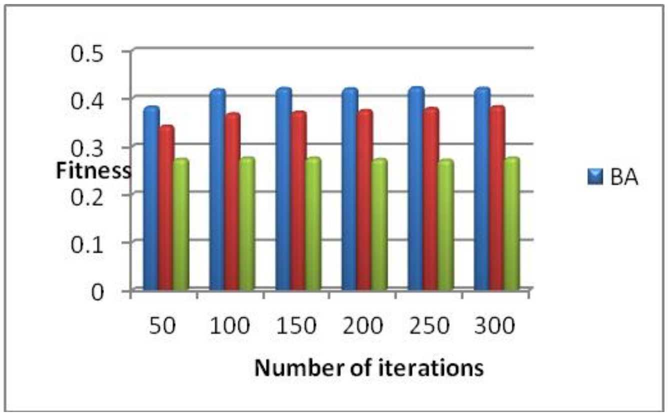

Figure 6: Number of iteration experiment

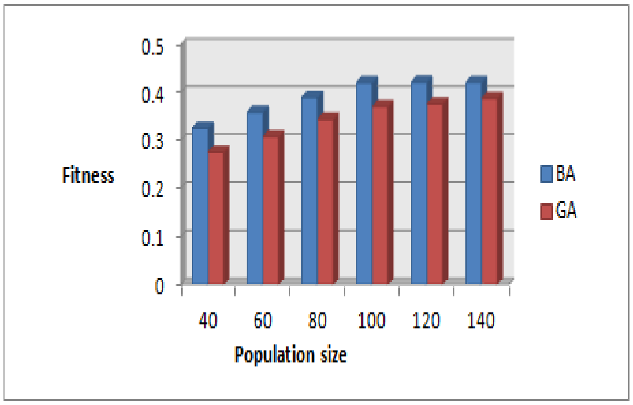

Figure 7: The population size experiment

The graph shown in figure 8 shows the execution time of the three algorithms, as it can be noticed the BA gives the highest execution time (more than the GA and K-means by 20% and 55% respectively), the GA comes after it and the K-means consumed the least time in algorithm execution. The extra time is expected but it is acceptable thanks to the increase in the accuracy.

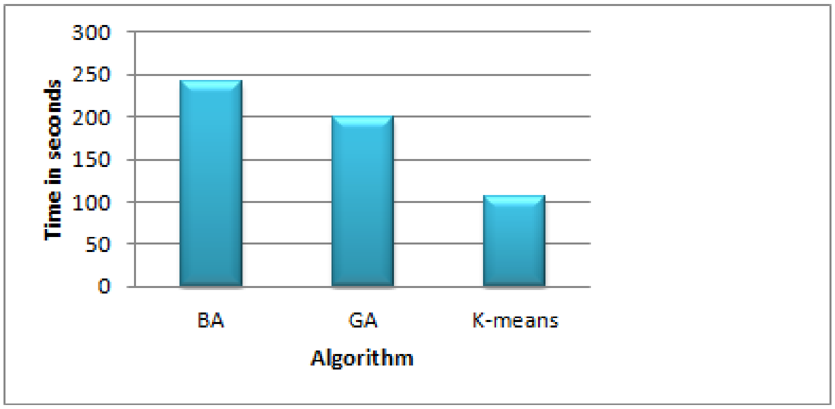

Figure 8: The execution time experiment

8 CONCLUSIONS AND FUTURE WORK

This paper has proposed a new technique in document clustering, the Bees algorithm. Cosine distance was used as a fitness function; The algorithm has proved its robustness via experiments on a data set of 818 documents against the Genetic Algorithm and K-means algorithm and shown its capability of getting solutions that best satisfy the fitness metric with an acceptable increase in the computation time over Genetic Algorithm and the K-means algorithm computation times.

Because of the algorithm ability to perform global and local search simultaneously, the algorithm avoids getting trapped into a local minima.

The Experiments has demonstrated that, In terms of fitness, the Bees Algorithm outperforms the Genetic Algorithm by 15% & the K-means by 50%. On the other hand, the Bees Algorithm consumes more time than the GA and the K-means by 20% and 55% respectively.

Our future plans will focus on hybridizing the Bees algorithm with other evolutionary algorithms and perform comparisons against other hybridized models. Moreover, we plan to apply experiments on the Bees Algorithm with larger corpus to test the algorithm scalability.

REFERENCES

[1] Pham, D.T. and Afify, A.A.: Clustering techniques and their applications in engineering. The Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science (2006)

[2] Nihal M. AbdelHamid, M.B. AbdelHalim, M.W. Fakhr: Document clustering using Bees Algorithm. International Conference of Information Technology, IEEE, Indonesia (March 2013)

[3] Goldberg D.E.: Genetic Algorithms-in Search, Optimization and Machine Learning. Addison- Wesley Publishing Company Inc., London (1989)

[4] K. Premalatha, A.M. Natarajan**:** Hybrid PSO and GA Models for Document Clustering. Int. J. Advance. Soft Comput. Appl. 2, 2074-8523 (2010)

[5] Shi, X.H., Liang Y.C., Lee H.P., Lu C. and Wang L.M., "An Improved GA and a Novel PSO-GA-Based Hybrid Algorithm", *Information Processing Letters*, 93, 5,255-261 (2005)

[6] Kennedy, J.; Eberhart, R.C. *Swarm Intelligence*. Morgan Kaufmann 1-55860-595-9 (2001)

[7] Mathur M, Karale SB, Priye S, Jayaraman VK and Kulkarni BD.: Ant Colony Approach to Continuous Function Optimization. Ind. Eng. Chem. Res. 3814-3822 (2000)

[8] Pham D.T., Ghanbarzadeh A, Koc E, Otri S, Rahim S and Zaidi M.: The Bees Algorithm. Technical Note, Manufacturing Engineering Centre, Cardiff University, UK (2005)

[9] Lukasz Machnik: Documents clustering method based on Ants Algorithms. 123 – 130 (2006)

[10] Bilchev G and Parmee IC. The Ant Colony Metaphor for Searching Continuous Design Spaces, Selected Papers from AISB Workshop on Evolutionary Computing. 25-39 (1995)

[11] Priya Vaijayanthi, Natarajan A M and Raja Murugados: Ants for Document Clustering, 1694-0814 (March 2012)

[12] Salton G.: Automatic Text Processing. Addison-Wesley (1989)

[13] Salton G., Wong A. and Yang C.,A.: Vector Space Model for Automatic Indexing. J. of Communications of the ACM, 18, 613–620 (1975)

[14] D.T. Pham, S. Otri, A. Afify, M. Mahmuddin and H. Al-Jabbouli:Data Clustering Using the Bees Algorithm, CIRP International Manufacturing Systems Seminar (2007).