Artificial Bee Colony based Approach for Web Information Retrieval

Dr. Hasanen S. Abdullah
Computer Science Department, University of Technology/Baghdad
Mustafa J. Hadi
Computer Science Department, University of Technology/Baghdad
Email:Mustafa_awad@yahoo.com

Received on: 2/6/2013 & Accepted on: 3/10/2013

ABSTRACT

With the tremendous growth of information in the web, the classic query processing approaches are unable to respond to queries in real time. The aim of this paper is to develop an innovative tool using swarm intelligence to address information retrieval in the context of response time and solution quality through cope with the complexity induced by that huge volume of information. In this paper, we will show that our proposed approach that use of Artificial Bee Colony (ABC) algorithm called MABC can be another alternative to palliate the complexity issue in terms of response time while it produces a solution quality is relatively convergent or even better. Experimental tests have been conducted on two well-known CACM and NPL collections. Both are different in size, CACM is small while NPL is relatively large. Numerical results exhibit the superiority and the benefit gained from using the MABC approach instead of the classic approaches.

Keywords: Web Information Retrieval; Swarm Intelligence; Artificial Bee Colony (ABC); MABC; Classic Approaches.

طريقة معتمدة على مستعمرة النحل الاصطناعية لاجل استرجاع المعلومات في الويب

الخلاصة

نتيجة للتزايد الهائل بالمعلومات الموجودة في الويب فان طرق معالجة الاستفسار التقليدية لم تعد قادرة على الاستجابة في الوقت الفعلي. الهدف من هذا البحث هو ايجاد طريقة بديلة باستخدام احدى خوارزميات التحشد الذكية لمعالجة استرجاع المعلومات في إطار الوقت اللازم لاجابة الاستفسار وجودة المعلومات من خلال التصدي للتعقيد الحاصل في الحجم الهائل في المعلومات. نحن سنبين في هذا البحث بان طريقتنا المقترحة التي تستخدم طريقة " مستعمرة النحل الاصطناعية" والتي تدعى "MABC" يمكن ان تكون بديلا عن الطرق التقليدية لتخفيف هذا التعقيد على صعيد الوقت اللازم لاستجابة الاستفسار بينما تنتج نوعية اجابة للمستخدم تكون متقاربة نسبيا مع الطرق التقليدية او حتى افضل منها . الاختبارات التجريبية

https://doi.org/10.30684/etj.32.5B.7
2412-0758/University of Technology-Iraq, Baghdad, Iraq

تمت على مدونتين معروفة جيدا احدهما صغيرة هي "CACM " والاخرى كبيرة نسبيا هي "NPL". النتائج الرقمية تبين التفوق والربح الحاصل للطريقة المقترحة على الطرق التقليدية.

INTRODUCTION

Information retrieval, namely IR for short has shown its great importance throughout the history of computer science. It has been widely used and has played a central role especially in large database management. In the Internet era, its interest is increasing more and more as it is considered as the core of many web applications. Now with the exponentially growth of information in the web, data sets are huge and very often not obvious to tackle with traditional approaches. In web information retrieval, the greater the number of documents to be searched, the more powerful approach required. In this context, artificial intelligence approaches and more precisely swarm intelligence approaches can be designed to efficiently browse the huge number of documents to find only the information needed by the user [1], [2], [3].

Swarm intelligence is an emerging area in the field of optimization and researchers have developed various algorithms by modeling the behaviors of different swarm of animals and insects such as ants, termites, bees, birds, and fishes. [4]

Artificial Bee Colony (ABC) algorithm is a recently introduced population-based meta-heuristic optimization technique inspired by the intelligent foraging behavior of honeybee swarms [5].

In this work, we developed innovative tool depending on Artificial Bee Colony (ABC) algorithm called MABC that indicate to Modified Artificial Bee Colony for improving the performance of information retrieval in the web context. The idea behind addressing the web information retrieval with swarm intelligence based approach is the pruning of the search space by exclusion of a large amount of information that is far from the best solutions and thus obtains high efficiency in query processing. The MABC algorithm is tested on CACM and NPL document collections and comparison with the traditional method is achieved.

INFORMATION RETRIEVAL

Information retrieval system handles and manages a collection of documents structured in an internal representation using an indexing process. It consists in finding a set of documents including information expressed in a query specifying user needs. The process involves a matching mechanism between the query and the documents of the collection. Therefore four important components are central in such process:

- The document which can be a text, a web page, an image or a video. A document is usually represented by a set of terms or keywords extracted from its source.
- The query which represents a need expressed by a user and specified in a formalism adopted by the system.
- The similarity function that measures the similarity between a document and a query.
- Two system evaluations are widely used: the precision which is the fraction of retrieved documents that are relevant and the recall which is the fraction of relevant documents that are retrieved.

In an IR system, an important step is the indexing process in which an internal organization of the documents, the terms and the queries is determined in order to access in an efficient way these components.

Besides the indexing process, the documents and the queries must be described according to a model. Many models for IR like the Boolean model, the vector space model (VSM) and the probabilistic model exist in the literature. The most widely used which is also appropriate for meta-heuristics is the vector space model. In this model, documents as well as queries are represented as vectors of weights. Each weight in the vector denotes the importance of the corresponding term in the document or in the query. The vector space is built during the indexing process and contains all the terms that the system encounters. Consider the following vector space:

$$(T1, t_2, t_3, \ldots, t_n)$$

Where ti is a term or keyword for i=*1 to n*. For each term, we consider a structure that contains all the documents that include the term. The weight of the term in the document is associated with the document in the list.

The whole collection of documents is represented by a vector containing all the documents. The structure is indexed by the number of the document.

$$C = (d_1, d_2, d_3, \ldots, d_m)$$

Each element of *C* points towards a list containing all the terms of the documents with their respective weight. The list is sorted according to the number of the term. The query is modeled exactly as a document.

The weight of a term in a document is computed using the expression *tf***idf* where *tf* is the term frequency in the document and *idf* is the inverted frequency computed usually as follows:

idf = log(m/df) where *m* represents the number of documents and *df* is the number of documents that contain the term. The component *tf* indicates the importance of the term for the document, while *idf* expresses the power of discrimination of this term. In this way, a term having a high value of *tf *idf* is at the same time important in the document and less frequent in the others.

The weight for a query is computed with the same manner. The similarity of a document *d* and a query *q* is then computed using the *cosine* formula:

$$f(d, q)=\frac{\sum i\,(a_i * b_i)}{\sqrt{\sum i\,(a_i)^2 * \sum i\,(b_i)^2}} \qquad \ldots(1)$$

Where a_i and b_i are the weight of term ti respectively in the document and in the query [1].

CLASSICAL SEARCH APPROACHES

Classical search approaches for information retrieval are exhaustive search and inverted index based search. With exhaustive search, the designed algorithm for IR

requires browsing the whole collection of documents and calculates the similarity between the document and the query. Generally, this algorithm is not considered even for reasonable size of corpuses because there is a smarter way to address this problem. The idea is to execute the above algorithm on the inverted index instead of the whole collection of documents. Only documents that have at least one common term with the query are consulted and this way the complexity of the algorithm is reduced at a phenomenal rate. However when we consider the web context, the inverted file remains scalable by containing several millions and more documents and may become untrea Table.

It is clear that the complexity in approach of the inverted index based search is more interesting than the one of the exhaustive search approach because fewer documents are considered in the search process. For both approaches the search process has a worst case complexity in *O (n*m)*, where *n* is the number of terms and *m* the number of documents. When *n* and *m* are reasonable, the inverted index technique is very efficient. However for an environment like the web where the number of documents and the number of keywords are prohibitive, the complexity is exponential because the parameters *n* and *m* express exponential magnitudes. This is the reason why it is important to find another alternative for addressing information retrieval in such context. Meta-heuristics enables to get a polynomial response time at a higher computation scale [1], [2], [3].

RELATED WORKS

A few authors and works have attempted to address web information retrieval and search engines with swarm intelligence approaches; we enumerate among them the authors: Drias, Mosteghanemi [1] designed a Bees Swarm Optimization (BSO) algorithm, namely BSO-IR for web information retrieval to explore the prohibitive number of documents to find the information needed by the user. Drias[2],[3] developed document search processes based on Particle Swarm Optimization (PSO) to improve the performance of Web information retrieval. Bou Ezzeddine [6] proposed an upgrading of a bee hive model for retrieving information from Web. Na´vrat, Kovacik [7] proposed a modified model of a bee hive for web search engine. Na´vrat, Jastrzembska´, Jeli´nek, Bou Ezzeddine, Rozinajova´ [8] presented a new approach for on-line web search engine inspired by the social behavior of honey bees.

ABC ALGORITHM

Artificial Bee Colony algorithm (ABC) was introduced by Karaboga [9] for function optimization.

Each solution (i.e., a position in the search space) represents a potential food patch and the solution quality corresponds to the food patch's quality. Agents (artificial bees) search and exploit the food sources in search space [10].

Figure (1) shows the pseudo-code of the Artificial Bee Colony algorithm [11].

ABC algorithm

1: Initialize the population of solutions x_{ij}.
2: Evaluate the population.
3: cycle = 1
4: Repeat
5: Produce new solutions v_{ij} for the employed bees and evaluate them.
6: Apply the greedy selection process for the employed bees.
7: Calculate the probability values p_i for the solutions x_{ij} .
8: Produce the new solutions v_{ij} for the onlookers from the solutions x_{ij} selected depending on p_i and evaluate them.
9: Apply the greedy selection process for the onlookers.
10: Determine the abandoned solution for the scout, if exists, and replace it with a new randomly produced solution x_{ij} .
11: Memorize the best solution achieved so far.
12: cycle = cycle + 1
13:Until cycle = I_{max}

Figure (1) ABC algorithm

The initial solutions are produced by the following equation:

$$x_{ij} = x_{min\ j} + rand\ [0, 1].\ (x_{max\ j} - x_{min\ j}) \quad ...(2)$$

Where $i = 1... N_S$, $j = 1...D$. N_S is the number of food sources and D is the number of optimization parameters. $x_{max\ j}$ and $x_{min\ j}$ respectively represent the upper and lower bounds for the dimension j.
The new solutions v_{ij} are produced by the following equation:

$$v_{ij} = x_{ij} + \phi_{ij}\ (x_{ij} - x_{kj}) \quad \ldots(3)$$

Where $k \in \{1, 2, \ldots, N_S\}$ and $j \in \{1, 2, \ldots, D\}$ are randomly chosen indexes. ϕ_{ij} is a random number between [-1, 1].
The probability p_i of selecting the food source i is determined by the following equation:

$$p_i = 0.9 \times \frac{f_i}{f_{max}} + 0.1 \quad ... (4)$$

where f_i is the fitness value of the current food source and f_{max} is the maximum fitness of food sources.
The scout randomly generates a new food source by the following equation:

$$v_{ij} = x_{min\ j} + rand\ [0, 1].\ (x_{max\ j} - x_{min\ j}) \quad ...(5)$$

The algorithm is stopped if the maximum number of iterations I_{max} is reached, or if the algorithm does not seem to converge in its initial phase [12],[13],[14],[15].

AUGMENTED IR SYSTEM

Information retrieval system which described in section 2 requires building of the documentary database that consist of inverted index and the retrieval model such as VSM completely far from the time of response to the queries. In this manner, any data structure can be offline augmented to the original documentary database in order to increase solution quality without significantly affect the response time although it may require more disk space. In this work, we will add a new data structure to original database that maintains information about pre-calculated similarity documents. This data structure includes the ranking calculation for each document in the whole documents collection with a certain threshold for disk space and CPU time issues. Let the collection of documents in the database be *C*, and the total number of documents in *C* is *m*. The same retrieval algorithm used online for the retrieve and rank the documents which correspond to a certain query will be now used offline among the documents themselves in the collection rather than between query and documents. The intended retrieval algorithm first computes relevance scores for all documents in *C* and then produce a ranking R_d of the documents based on the relevance scores, i.e.,

$$R_d: <d_1^d , d_2^d ,\ldots, d_m^d>$$

Where $d_1^d \in C$ is the most relevant document to document d and $d_m^d \in C$ is the most irrelevant document to document *d*.

This data structure or the ranked list in other words, will be used later in the search using Artificial Bee Colony algorithm in the query processing time to make the search has something seem guided rather than to be absolutely stochastic. To difference this offline ranked list of documents from the online ranked list that is built during query processing, we will name this ranked list as "Nearest-List" that means the documents that are nearest of a particular document.

MABC ALGORITHM

Depending on the ABC algorithm in Figure (1) described in section 5, and in order to this algorithm suits the information retrieval environment, we modified the original ABC algorithm to MABC algorithm. The Figure (2) below briefly describes the pseudo-code of our proposed algorithm.

MABC algorithm

1: Initialize the population of solutions x_i from the collection.
2: Evaluate the population and memorize the best solution with its global Nearest-List.
3: cycle = 1
4: Repeat
5: Produce new solutions v_i for the employed bees by local Nearest-List and evaluate them.
6: Apply the greedy selection process for the employed bees.
7: Calculate the probability values p_i for the solutions x_i by Equation (4).
8: Produce the new solutions v_i for the onlookers bees by global Nearest-List
From the solutions x_i selected depending on p_i and evaluate them.
9: Determine the abandoned solution for the scout, if exists, and replace it with
a new randomly produced solution x_i from the collection.
10: Apply the greedy selection process for the onlookers.
11: Memorize the best solution (achieved so far) with its global Nearest-List.
12: cycle = cycle + 1
13:Until cycle = I_{max}

Figure (2) MABC algorithm.

Local and global Nearest-List represents a sequence cutoff of all food sources (documents) that are nearest to a certain food source. Local refers to the nearest to current food source while global refers to the nearest to best food source.

EXPERIMENTAL RESULTS

The experimental evaluations are in terms of time and quality with compared to the classic one which refers to approach of the inverted index based search. Indeed, there is no need to compare our approach with exhaustive search approach which is explicitly less efficiency from the second approach. Our system is experimented on two different document collections CACM (3204 documents, 64 queries with 52 judgments) and NPL (11429 documents, 93 queries with 93 judgments). These two collections are well-known and used in many research works for evaluating IR systems. Tables (1) and (2) below show the average latency to evaluate the efficiency and also the average of two effectiveness measures precision and recall at rank 10 with their 11-point interpolated recall-precision curve showed in Figures (3) and (4) for all queries in CACM and NPL collections. In our work we used a term weighting approach called “Best fully weighted system” described in [16].
The comparison with the classic approach and then the gain average in performance are also presented in Tables (1 and 2).

Table (1) Average performance evaluation with CACM collection for all queries.

Algorithm	Classic	MABC
Avg. of visited docs for each query out of 3204 docs	1240	784
No. of zero relevant answers	5	3
No. of full relevant answers	0	1
Avg. latency (in sec.)	0.781	0.6596
Avg. precision	0.3308	0.3462
Avg. recall	0.3117	0.3226
Avg. of gain in time (in sec.)	0	0.1214
Avg. of gain in Precision	0	0.0154
Avg. of gain in Recall	0	0.0109

Table (2) Average performance evaluation with NPL collection for all queries.

Algorithm	Classic	MABC
Avg. of visited docs for each query out of 11429 docs	2702	1186
No. of zero relevant answers	12	12
No. of full relevant answers	1	1
Avg. latency (in sec.)	1.5884	1.0728
Avg. precision	0.2624	0.272
Avg. recall	0.1706	0.1816
Avg. of gain in time (in sec.)	0	0.5156
Avg. of gain in Precision	0	0.0097
Avg. of gain in Recall	0	0.011

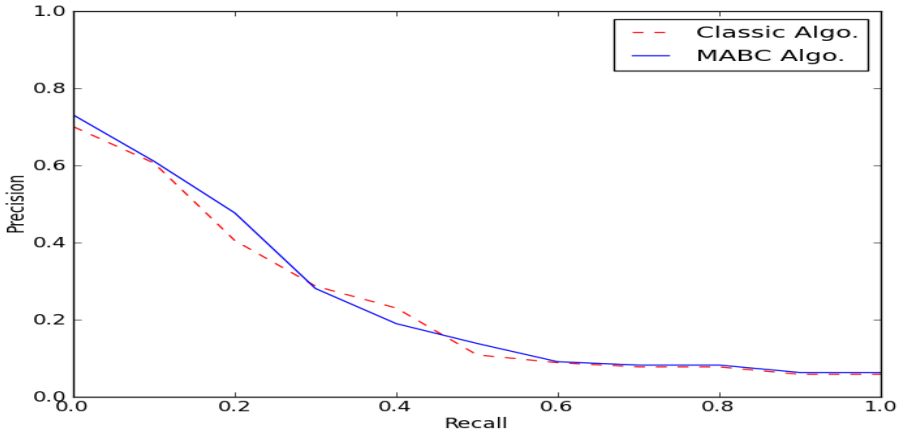

Figure (3) Average recall-precision curve for all queries in CACM collection.

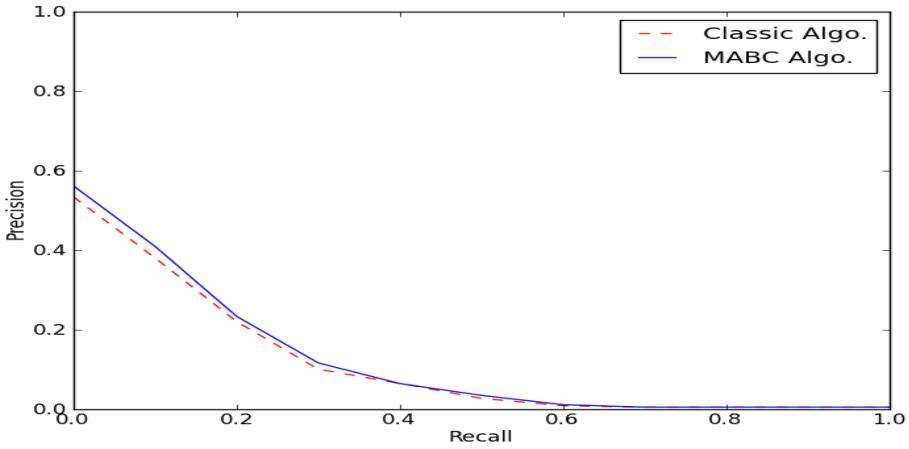

Figure (4) Average recall-precision curve for all queries In NPL collection.

CONCLUSIONS

In this paper, we developed innovative tool using swarm intelligence for cope with the complexity induced by the huge volume of information on web environment. The bio-inspired approach Artificial Bee Colony (ABC) can be another alternative to the classic approaches in order to improve query processing efficiency. Experimental tests have been conducted on two well-known CACM and NPL collections. Both are different in size, CACM is small while NPL is relatively large. We can enumerate the conclusions of our work as follows:

1: We introduce a study on how to adapt a swarm intelligence technique to a large scale IR problem and its comparison with the classic approach.

2: Numerical results of our proposed algorithm MABC show the benefit gained from using such approach instead of the classic one, and thus exhibit the superiority of our system on previous classical works in terms of response time while it produces a solution quality is relatively convergent or even better.

3: Numerical results show also that the benefit gained in time increases if size of collection is larger and thus prove that our system is more suitable to large scale IR and especially for environment as the web.

4: The classic approach which is completely depending on the inverted index in search process suffer from its severe relation with the similarity calculations and term-weighting schemes of the documents and queries, thus the solutions quality will be hardly determined due to the constrained competition among the documents toward the appearance to the users on the basis of their queries. In contrast, the solutions quality of stochastic search used in our system is softly determined and not related completely to the similarity or term-weighting schemes and thus it can get documents in a region that is hard to access because it surrounded by barren regions in the search space.

FUTURE WORK

Our system does not use the inverted index in search process. However, it uses the inverted index to calculate the term weighting for queries in the response time. This time slice has calculated in our system and it in itself considered as wasted time added to the search time. We intent in the future work to develop a tool in some way to deduction the exact or approximate weight for each term in the query directly without using the inverted index and thus this will decrease more the response time .

Another future work, we plan to hybridize MABC with another bio-inspired approach in order to better address web information retrieval through consider the advantages for both to improve more of the solution quality.

REFERENCES

[1]. Drias, H. H. Mosteghanemi: "Bees Swarm Optimization based Approach for Web Information Retrieval", IEEE/WIC/ ACM International Conference on Web Intelligence and Intelligent Agent Technology, 2010.

[2]. Drias: H. "Web Information Retrieval using Particle Swarm Optimization based Approaches", IEEE/ WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, 2011.

[3]. Drias: H. "Parallel Swarm Optimization for Web Information Retrieval", IEEE Third World Congress on Nature and Biologically Inspired Computing, 2011.

[4]. Dervis Karaboga, Bahriye Akay" A survey: algorithms simulating bee swarm intelligence", Artif Intell Rev (2009) 31:61–85 DOI 10.1007/s10462-009-9127-4, Springer Science+Business Media B.V. 2009

[5].Fahad S. Abu-Mouti, Mohamed E. El-Hawary:" Overview of Artificial Bee Colony (ABC) algorithm and its applications", IEEE International, Systems Conference (SysCon), 2012

[6]. Bou Ezzeddine, A. "Web information retrieval inspired by social insect behavior". Information Sciences and Technologies Bulletin of the ACM Slovakia, Vol. 3, No. 1 (2011) 93-100

[7]. Pavol Navrat, Martin Kovacik:" Web Search Engine as a Bee Hive". IEEE/ WIC/ACM International Conference on Web Intelligence,2006

[8]. Pavol Navrat, Lucia Jastrzembska, Tomas Jelinek, Anna Bou Ezzeddine, Viera Rozinajova:" ," Exploring Social Behavior of Honey Bees Searching on the Web", IEEE/ WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, 2007.

[9]. Karaboga, D.: An idea based on honey bee swarm for numerical optimization. Technical Report TR06, Computer Engineering, Department, Erciyes University, Turkey, 2005.

[10]. Bijaya Ketan Panigrahi,Yuhui Shi, and Meng-Hiot Lim:" Handbook of Swarm Intelligence: Concepts, Principles and Applications", Springer-Verlag Berlin Heidelberg,2011.

[11]. Bahriye Akay, Dervis Karaboga," A comparative study of Artificial Bee Colony algorithm" Department of Computer Engineering, Erciyes University, 38039 Melikgazi, Kayseri, Turkey, journalhomepage, 2009 www.elsevier.com/locate/amc.

[12]. Bahriye Akay, Dervis Karaboga," A modified Artificial Bee Colony algorithm for real-parameter optimization" Department of Computer Engineering, Erciyes University ,38039Melikgazi, Kayseri,Turkey,journalhomepage,2010 www.elsevier.com/locate/ins

[13].Ali Hadidi, Sina Kazemzadeh Azad, Saeid Kazemzadeh Azad:" Structural optimization using artificial bee colony algorithm", 2^{nd} International Conference on Engineering Optimization , Lisbon, Portugal, Department of Civil Engineering, University of Tabriz, Tabriz, Iran,2010.

[14]. Mahmoud Maher Jahjouh," Design Optimization of Reinforced Concrete Frames using Artificial Bee Colony Algorithm",The Islamic University of Gaza, High Studies Deanery, Faculty of Engineering, Civil Engineering Department, Design and Rehabilitation of Structures, MS.c thesis in Civil Engineering – Design and Rehabilitation of Structures,2012.

[15]. Ganga Reddy Tankasala," Artificial Bee Colony Optimization for Economic Load Dispatch of a Modern Power system", International Journal of Scientific & Engineering Research, ISSN 2229-5518, Volume 3, Issue 1, January-2012.

[16] H. Kang, K. Choi:" Two-Level Document Ranking Using Mutual Information in Natural Language Information Retrieval". Information Processing and Management. Vol. 33, No. 3. pp. 289-306. 1997.