A Novel PSO Methodology for Web Documents Retrieval

Ramya C
Department of Computer Science & Engineering
U. B. D. T College of Engineering, Davangere University
Karnataka, India

ABSTRACT: *This paper focuses on retrieval of web documents with improved response time and similarity using particle swarm optimization (PSO) technique. Since the nature of the web data is distributed, volatile and uncertain, an accurate and speedy access is required. Hence a novel approach on evolutionary bio-inspired Swarm Intelligence techniques to optimize search process in Web Information Retrieval systems is proposed and developed. Here, we propose a novel algorithm using basic PSO technique which works on both small CACM and huge RCV1 collections. We apply this on the pre-processed documents to retrieve most similar documents with a very less response time. This paper also reveals a comparative study with the existing method.*

Keywords: Particle Swarm Optimization, Web Information Retrieval (WIR), Pre-processing of Documents, Indexing

Received: 18 May 2019, Revised 28 July 2019, Accepted 5 August 2019

DOI: 10.6025/jcl/2019/10/3/67-75

1. Introduction

As the Internet is ubiquitous, it's become easy for the users to access required information on the World Wide Web (WWW). It leads to tremendous growth of web documents on the WWW. Though Search engines provide required information to the users based on their queries, many times users are not satisfied with the retrieval accuracy and speed of search engines. Since the web data stored is distributed [9], search engines sometimes fail to retrieve more relevant documents for the user queries though they are available in system. Rather they are fetching irrelevant web documents. So there is a need for the optimization of this search process in WIR system. Several techniques such as Genetic Algorithms [13], Simulated Annealing, Neural networks, etc. are used to optimize the search process efficiently. Though these techniques are efficient, there is still a scope to reduce query response time and to higher the similarity of the best found documents.

SI is best suited for optimization problems like TSP, quadratic assignment, graph colouring, optimization, network routing, cluster finding, job scheduling, search engines and load balancing etc.. Swarm intelligence has been applied successfully to a wide variety of search and optimization problems. The main strength of PSO as an algorithm is its fast convergence. So we propose a novel PSO approach to reduce response time of the system and improve similarity (resemblance) of document and

query hence optimising the WIR process. Proposed method will handle both small and huge collections of web documents. For comparing performance of the proposed work, we considered another work carried out as given in paper [1] as the existing method.

The rest of the paper is organized as follows. In section 2, we discuss the related research in application of swarm intelligence to the problems what web information retrieval is facing with. The proposed PSO approach and its component details are presented in section 3. In section 4, we give the results of our experiments. Finally, we conclude and discuss the perspectives in section 5.

2. Related Work

Swarm Intelligence are relatively new areas where research is going for smooth information retrieval. The survey on how swarm technology has been applied to solve the optimization problems of WIR process is briefly presented in this section. Habiba Drias et al. [1] showed the designing of two novel PSO algorithms as search techniques for information retrieval on the web. Here the PSO is parallelized by executing the code independently and in parallel to each particle said in the algorithm. Due to parallelism, the PSO threads will return a high similarity of the best found document with a less runtime as the best solution. It is shown through experimental results that parallel PSO outperforms all the other heuristic search methods [10, 11, 12]. However when it comes to large collections of documents PSO achieves little similarity compared to classical exact algorithm [19] but provides significant response time than it.

The bee hive model inspired by bee swarm behavior for retrieving information from the web is proposed by Anna Bou Ezzeddine et al. [2] is an upgraded one. An adapted model in the form of bees is used to trace the story that is developing on the Web on-line. Peiyu Liu, Zhenfang Zhu and Lina Zhao [3] introduced this kind of clustering algorithm based on the ant heap which is not pre-specified the number of clusters, the arbitrary shape of cluster is constructed. This algorithm is the same with pile, which can be picked up or down again like a single object and form a new cluster again.

Dr. Hasanen S. Abdullah and Mustafa J. Hadi [4] used Artificial Bee Colony (ABC) algorithm with the aim of addressing information retrieval with the huge volume of information in terms of response time and good solution quality. The comparison with classic approach in terms of response time and document quality showed explicitly less efficiency of the classic approach. However this work is inspired by [5] where the adaptation of heuristic search techniques to large scale IR and their comparison with classical approaches can be seen.

A different method for web mining particularly in ranking web pages was developed by G. Anuradha and G. Lavanya Devi [22] using the same Artificial Bee Colony (ABC) approach. It considers users interest, total web site linkage and growth analysis rate are used to assign rank to the web pages. Proposed ABC approach for ranking web pages is implemented and tested on real datasets. The goal of this algorithm is to assign rank for web pages based on Users Interest, Total sites linking to the web page (Page Ranking), Growth Analysis rate.

PSO can be hybridized with Genetic Algorithm to get better results, Priya I. Borkar and Leena H. Patil et al. [6] presented a model of hybrid Particle Swarm Optimization (HGAPSO) to produce the new keywords that are related to the user search. The Jaccard similarity function is used to find the fitness value of each document. Fast and high-quality document clustering algorithms play an important role in effectively navigating, summarizing, and organizing information. Xiaohui Cui, Thomas E. Potok and Paul Palathingal [7] presented a Particle Swarm Optimization (PSO) document clustering algorithm. Contrary to the localized searching of the K-means algorithm, the PSO clustering algorithm performs a globalized search in the entire solution space. The hybrid PSO algorithm combines the ability of globalized searching of the PSO algorithm and the fast convergence of the K-means algorithm and avoids the drawback of both algorithms.

3. Proposed Methodology

The Proposed architecture is as shown in Figure 1. The diagram shows three components: input, WIR process and output. WIR process runs in the background, uses documents collection and the user query as input and retrieves the most relevant documents at the top as output. When the retrieval system is on-line, it is possible for the user to change his request during one search session in the light of a sample retrieval, thereby, it is hoped, improving the subsequent retrieval run. Such a procedure is commonly referred to as *feedback.*

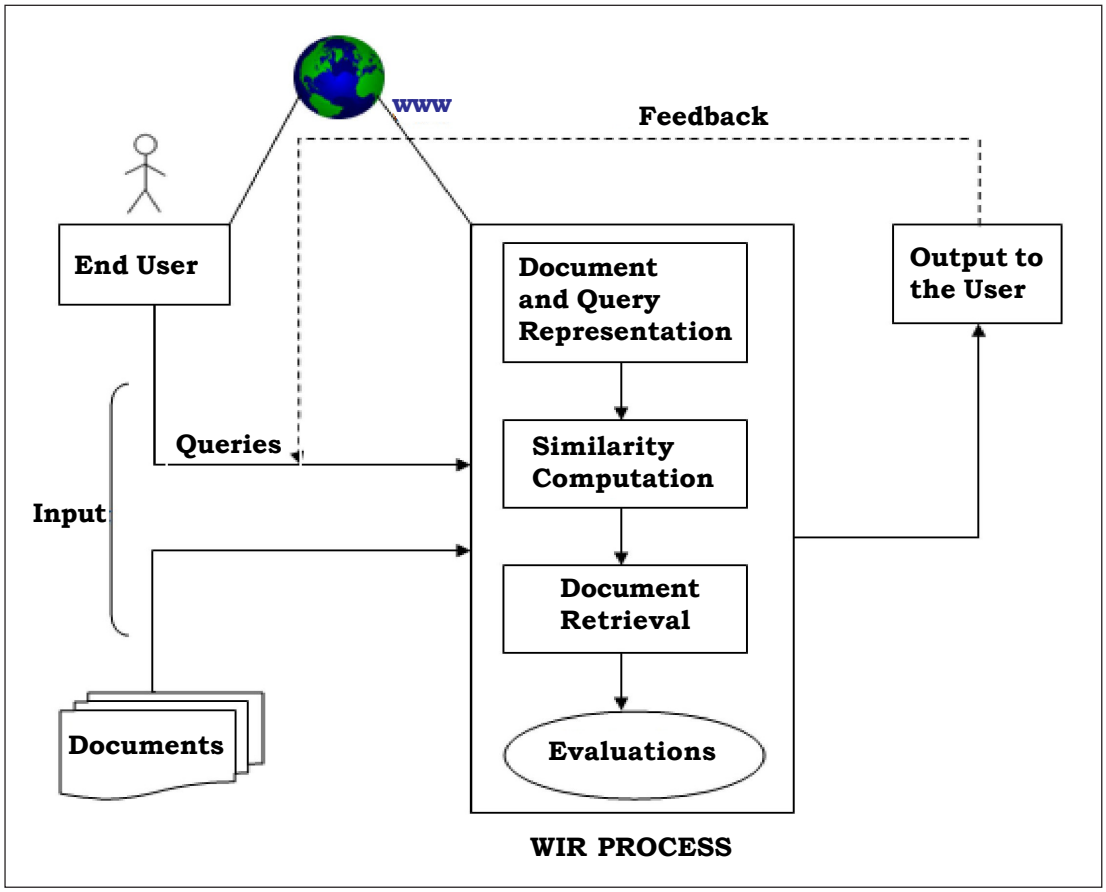

Figure 1. Proposed Architecture

3.1 Preprocessing of Documents

The initial corpuses of the documents are formed by indexing so that these documents are accessed in an efficient manner. We are using Lucene for indexing the documents which is a high-performance, full-featured text search engine library written entirely in Java. Non-verbal words are used for the similarity measurements. To get the reduced indexing file size removal of stop words is done as stop words are not useful in searching. Then clustering of documents is done as mentioned in algorithm steps 5-7. This is called preprocessing of documents.

3.2 Model and Similarity Functions used

The vector space model is used to represent each document and the query as vectors. In this model a similarity coefficient (SC) that measures the similarity between a document and a query can be computed.

The sum of following three similarity functions is used to measure the degree of resemblance between a document and a query:

$$f(d, q) = \Sigma_i(a_i * b_i) / (\Sigma_i(a_i)^2 * \Sigma_i(b_i)^2)^{1/2} \, (Cosine)$$

$$f(d, q) = 2\,\Sigma_i(a_i * b_i) / (\Sigma_i(a_i)^2 + \Sigma_i(b_i)^2) \, (Dice)$$

$$f(d, q) = \Sigma_i(a_i * b_i) / (\Sigma_i(a_i)^2 + \Sigma_i(b_i)^2 - \Sigma_i(a_i * b_i)) \, (Jaccard)$$

3.3 Novel PSO Algortihm

After preprocessing we are applying proposed PSO algorithm on web documents for retrieval purpose. The indexed documents and the query is considered as input. The steps of proposed PSO algorithm is as shown below.

Steps:

1. Initialize the indexing for the documents.
2. Assign docs to indexwriter

$$indexwriter_i = docs_i$$

3. Compute maximum no of terms in each and every an index.

$$Maxterms_i = indexwriter_i .countterms()$$

4. Repeat the steps 2 and 3, $\forall i = 1, 2,...n$
5. Initialize the Clustering.
6. Compute the Cluster,

$$Cluster_j = \begin{cases} Maxterms_j, & Maxterms_j < Maxterms_{j+1} \\ Maxterms_{j+1}, & Otherwise \end{cases}$$

7. Repeat the step 6 $\forall i = 1, 2,...n$
8. Initialize the particles by assign Clusters, for each clusters i to m.
9. Compute the velocity

$$\boldsymbol{randpart = randval, 0} < randval < \boldsymbol{((insf + 1) * 2)/2}$$

$$\boldsymbol{randglob = randval, 0} < randval < \boldsymbol{((insf + 1) * 2)/2}$$

$$vel_{bc} = insf * vel_{bc} + randpart * (part_{bc} - newvel_{bc}) + randglob * (bss - newvel_{bc})$$

$$newvel_{bc} = (newvel_{bc} + vel_{bc}) \; mod \; (totdocscount) + 1$$

10. Update the best solution for particle best cluster.

$$newvel_{bc} > f(part_{bc}) \rightarrow part_{bc} = newvel_{bc}$$

$$\boldsymbol{f(part_{bc}) > f(bss) \rightarrow bss = part_{bc}}$$

11. Repeat the steps 9 and 10 for every particles.

Where,

- ***i, j*** is the looping counters.
- ***n, m*** is the maximum document counts.
- ***indexwriter*** is adding all the documents to form an indexing.
- **maxterms** is collecting maximum no of terms in each and every an index.
- ***cluster*** is cluster for maxterms based on maximum terms in an index.
- ***bc*** is the best cluster for maxterms.
- ***randpart, randglob*** is the confidence coefficient for unsigned random numbers.
- ***insf*** is an inertial factor.
- ***part*** is the particle.
- ***vel*** is velocity.
- ***newvel*** is new velocity.
- ***bss*** is the best solution for swarm.

- ***part*[*bc*]** is the best solution for particle best cluster.

- ***f*** is the fitness function calculated by the similarity coefficient function.

4. Experimental Results and Discussion

The proposed algorithm has been implemented in Java. Apache Tomcat Server 7.0.33 is used for HTTP Protocols. Mysql is for Backend Process. And Xampp is used which helps to execute the dynamic application. The empirical parameters such as number of iterations and number of particles are set by experiments.

The two collections of documents are used. Firstly CACM, considered to besmall collection which is of HTML documents containing article abstracts published in ACM journal between 1958 to 1979. Another one is RCV1, considered as a huge collection which is of XML documents representing archives published in Reuters. Table 1 shows the characteristics of datasets used.

Data Sets	No. of Documents	No. of Terms	Average Document Size (Bytes)
CACM	3,204	6,468	2K
RCV1	8,04,414	47,236	2K

Table 1. Characteristics of Datasets

IRS should rapidly find out good quality results to the queries based on keywords. Here the term 'rapidly' implies 'response time' of the IRS and the term 'quality' implies most 'similar' documents to the query. So we tried to improve response time of the system as well as similarity of the documents to the query. Table 2 shows obtained similarity, response time and the ratio of similarity to the response time for the four different queries. This ratio is used to analyse the efficiency of the algorithm. Suggested document count is the number of documents retrieved. So in Table 2 we can observe the considerable reduction in response time keeping similarity relatively stable.

Algorithm	Query	Suggested Document Count	Similarity	Response Time(ms)	Similarity/Response Time
Existing	1	69	15.29	172	0.0889
	2	27	7.14	93	0.0768
	3	13	5.5	69	0.0802
	4	193	56.7	284	0.1997
Proposed	1	69	15.29	81	0.1888
	2	27	7.14	15	0.476
	3	13	5.5	5	1.1065
	4	193	56.7	219	0.259

Table 2. Proposed v/s existing method on CACM Collection

The effect of both the algorithms on huge collection RCV1 data sets is shown in Table 3. The results are tabulated for 4 queries. Though the collection is huge, we can observe the amount of reduction in response time keeping same similarity values and hence superiority of the proposed over existing algorithm can be noticed. Figure 2-5 show the comparison of both proposed and existing algorithm performances on both CACM and RCV1 collections.

Algorithm	Query	Suggested Document Count	Similarity	Response Time(ms)	Similarity/Response Time
Existing	1	13328	1290.38	29062	0.0444
	2	6832	563.76	70332	0.008
	3	9517	1392.56	83593	0.0167
	4	163	12.55	2704	0.0046
Proposed	1	13328	1290.38	28778	0.0448
	2	6832	563.76	14917	0.0378
	3	9517	1392.56	22469	0.062
	4	163	12.55	389	0.0323

Table 3. Proposed v/s existing method on RCV1 Collection

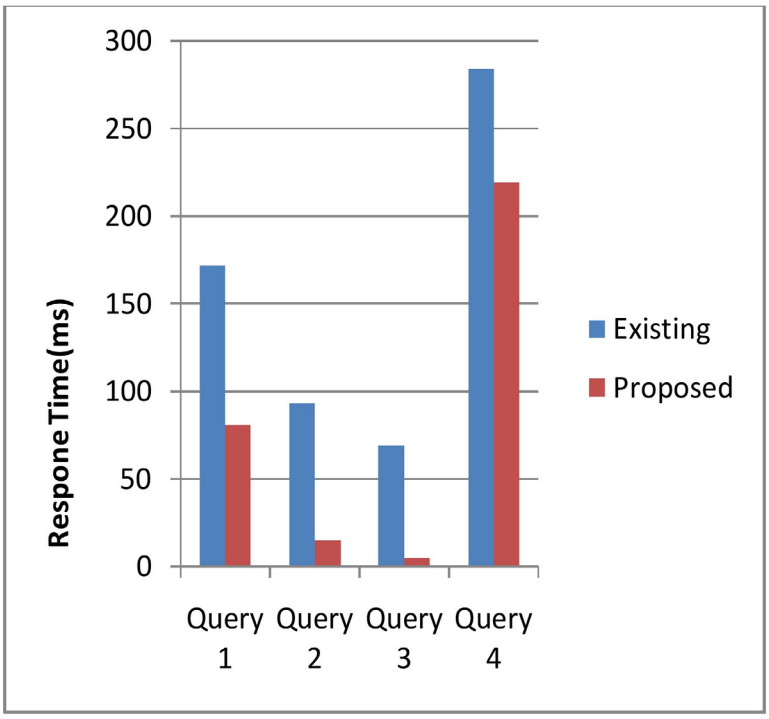

Figure 2. Response time of proposed and existing IRSs for CACM datasets

5. Conclusion and Future Work

In this paper, we have developed a novel PSO optimization method for document retrieval. The approach has a strong capability of significantly reducing response time keeping good similarity. Compared with existing method, our approach has been testified to possess superior performance in terms of response time, stability and robustness. The results demonstrate that our method is well suited to handle both small and huge collections. It is implemented in such a way that one can use it as an application for document retrieval. It is desirable to further apply PSO for providing high similarity of the best found document. The future work includes the studies on how to extend PSO to handle those IR optimization problems, such as personalizing the source selection, distributed information retrieval and so on.

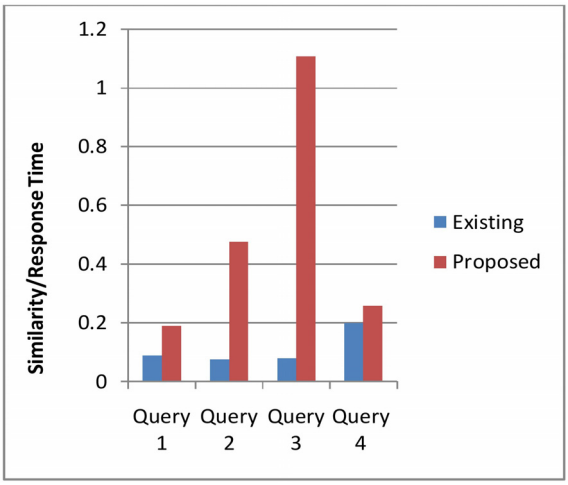

Figure 3. Similarity/time ratio of proposed and existing IRSs for CACM datasets

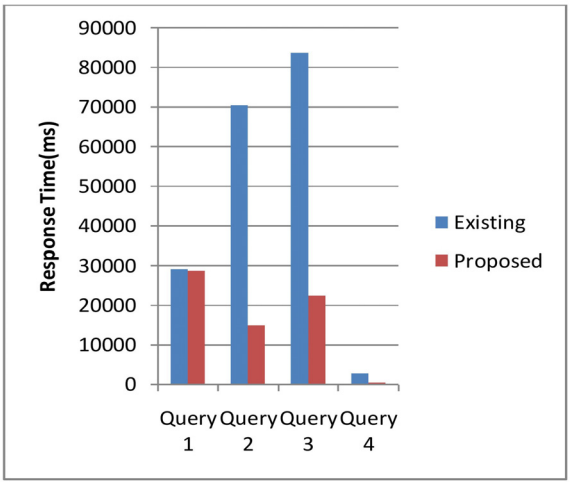

Figure 4. Response time of proposed and existing IRSs for RCV1 Collection

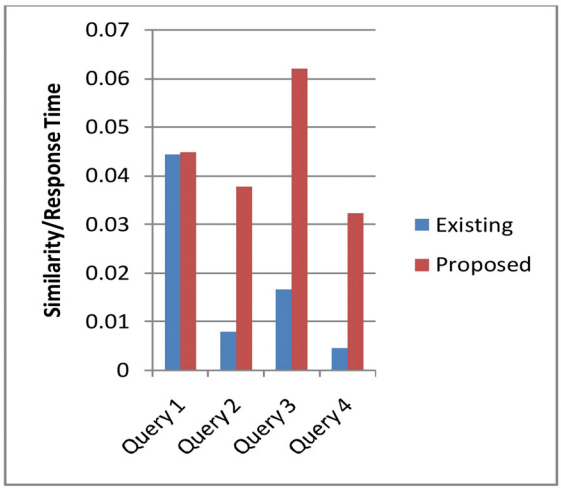

Figure 5. Similarity/time ratio of proposed and existing IRSs for RCV1 Collection

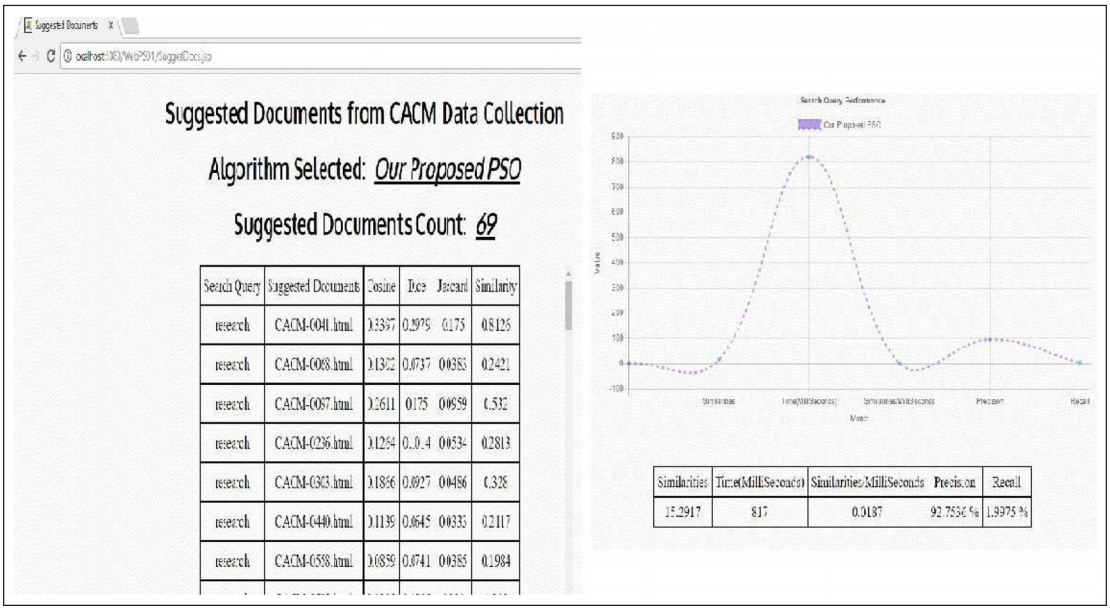

Figure 6. Snapshot showing output of proposed algorithm

References

[1] Drias, Habiba. (2011). Parallel Swarm Optimization for Web Information Retrieval, *In*: Proceedings of Third World Congress on Nature and Biologically Inspired computing, 249-254.

[2] Anna Bou Ezzeddine. (2011). Web information retrieval inspired by social insect behavior. *Information Sciences and Technologies Bulletin of the ACM Slovakia*, 3(1) 93-100.

[3] Liu, Peiyu., Zhu, Zhenfang., Zhao, Lina. (2009). Research on Information Retrieval System Based on Ant Clustering Algorithm. *Journal of Software*, 4(9) 1032-1036.

[4] Hasanen, S., Abdullah., Mustafa, J., Hadi. (2014). Artificial Bee Colony based Approach for Web Information Retrieval. *Engineering and Technology Journal*, 32(5) 899-909.

[5] Drias, Habiba., Mosteghanemi, Hadia. (2010). Bees Swarm Optimization based Approach for Web Information Retrieval, *In*: Proceedings of IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, 6-13.

[6] Priya, I., Borkar, Leena, H., Patil. (2013). Web Information Retrieval Using Genetic Algorithm-Particle Swarm Optimization, *International Journal of Future Computer and Communication*, 2(6).

[7] Cui, Xiaohui., Thomas, E., Potok, Palathingal, Paul. (2005). Document Clustering using Particle Swarm Optimization, *In*: Proceedings of IEEE Swarm Intelligence Symposium, 185-191.

[8] Navrat, Pavol., Anna Bou Ezzeddine. (2010). Bee Hive at Work: Following a Developing Story on the Web, Artificial Intelligence in Theory and Practice III, 331, Springer, 187-196.

[9] Kohli, Shruti., Gupta, Ankit. (2014). A Survey on Web Information Retrieval Inside Fuzzy Framework, *In*: Proceedings of Third International Conference on Soft Computing for Problem Solving. *Advances in Intelligent Systems and Computing*, 433-445.

[10] Yates, Baeza., R., Neto, Ribiero. B. (1999). Modern Information Retrieval, Addison Wesley Longman Publishing Co. Inc.

[11] Manning, C. D., Raghavan, P., Schutze, H. (2008). Introduction to Information Retrieval, Cambridge University Press.

[12] Kennedy, J., Eberhart, R. C. (1995). Particle Swarm Optimization, *In*: Proceedings of the IEEE International Conference On Neural Networks, Piscataway, NJ, 1942-1948.

[13] Pathak, P., Gordon, M., Fan, W. (2000). Effective Information Retrieval using Genetic Algorithms based Matching Functions Adaptation, 33rd IEEE HICSS.

[14] Hsinchun, C. (1995). Machine Learning for Information Retrieval: Neural Networks, Symbolic Learning and Genetic Algorithm, *Journal of the American Society for Information Science*, 194-216.

[15] Abraham, Ajith., Guo, He., Liu, Hongbo. (2006). Swarm Intelligence: Foundations, Perspectives and Applications, Studies in Computational Intelligence (SCI) 26, 3–25.

[16] Van Ast, J., Babuska, R., De Schutter, B. (2008). Particle swarms in optimization and control, *In*: Proceedings of the 17th IFAC World Congress, Seoul, Korea, 5131–5136, (July).

[17] Teodorovic, Dusan. (2006). Bee Colony Optimization: Principle and Applications, Eighth IEEE Seminar on Neural Network Applications in Electrical Engineering, NEUREL 2006, (September).

[18] Dorigo, Marco., Birattari, Mauro., Stutzle, Thomas. (2006). Ant Colony Optimization: Artificial Ants as a Computation Intelligence Technique, IEEE Computational Intelligence Magazine, 28-39, (November).

[19] Salton, G., Buckley, C. (1988). Term weighting approaches in automatic text retrieval, *Information Processing and Management*, 513-523.

[20] Bratton, Dan., Blackwell, Tim. (2008). A Simplified Recombinant PSO, *Journal of Artificial Evolution and Applications*, Hindawi Publishing Corporation, 1-10.

[21] Karol, Stuti., Mangat, Veenu. (2012). Survey on Particle Swarm Optimization based Web Mining, *Journal of Information and Operations Management*, 3(1) 273-276.

[22] Anuradha, G., Devi, Lavanya. G. (2014). Artificial Bee Colony (ABC) Approach for Ranking Web Pages, *International Journal of Computer Applications* (0975 – 8887) 99(1), (August).

[23] Rocchio, J. (1971). Relevance Feedback in Information Retrieval, In G. Salton, editor, The Smart Retrieval System: Experiments in Automatic Document Processing, Prentice-Hall, Englewood Cliffs, NJ, 313–323.

[24] Christopher, D. (2009). Manning, Prabhakar Raghavan and Hinrich Schütze, An Introduction to Information Retrieval, Cambridge University Press.

[25] Grosan, Crina., Abraham, Ajith., Chis, Monica. (2006). Swarm Intelligence in Data Mining, Studies in Computational Intelligence (SCI) 34, 1–20.

[26] Kennedy, J., Eberhart, R. C. (1995). Particle Swarm Optimization, *In*: Proceedings of the IEEE International Conference on Neural Networks, Perth, Australia, 1942–1948.