The Effect of Hybrid Crossover Technique on Enhancing Recall and Precision in Information Retrieval

Ammar Al-Dallal

***Abstract*— Several techniques are proposed to retrieve the most relevant HTML documents to user query. Among these techniques is the genetic algorithm which iteratively creates several generations using selection, crossover and mutation before producing the final result. In this paper, a new hybrid crossover technique is proposed to enhance the quality of the retrieved results. This technique is applied to HTML documents and evaluated using recall, precision and recall-precision measures. Its performance is compared to three well known techniques of crossover. The results show high improvement in the quality of the retrieved documents in terms of these measures.**

***Index Terms*— genetic algorithm; HTML documents; hybrid crossover; information retrieval**

I. INTRODUCTION

The study of IR techniques has increased since the advent of the World Wide Web, but still web users encounter two problems when trying to retrieve useful information. One of them is that many of the highly ranked retrieved documents are not related to the user query. On the other hand, there still many related documents which are not retrieved [22]. For this reason, many paradigms and models have been developed to solve the IR problem. One of these is genetic algorithm (GA). This technique uses the principles of selection and evolution to produce several solutions to a given problem.

There are many approaches investigate GA engine that can solve web search problem [5][12][13][17][20][22]. In this type of approaches, GA generates a population which is a group of individuals called chromosomes and each chromosome consists of a set of genes selected randomly. These genes in the proposed system represent an index to an HTML document in the search space. The individuals in the population are then evaluated using fitness function which is provided by the programmer and gives the individual a score based on how well it reflects relativity to the user query.

Next generations are re-produced from previous ones using selection, crossover and mutation. The selection of individuals is controlled by the fitness function. This continues until a suitable solution has been found or a certain number of generations have been passed.

Ammar Al-Dallal is with the Computer Engineering department, Ahlia University, Manama, Kingdom of Bahrain (e-mail: aaldallal@ahlia.edu.bh)

The rest of this paper is organized as follows: Section 2 lists the related work. Section 3 explains in details the proposed hybrid crossover technique which will overcome many of the drawbacks of the existing crossover techniques. The description of the document set used is provided in section 4. Section 5 describes the measures applied to evaluate the proposed hybrid technique. Section 6 represents and explains the achieved results. Last section concludes this work by highlighting ideas to enhance this work.

II. RELATED WORK

This paper focuses on the crossover part of the GA process. Once the individuals are selected using the selection operator, they are ready for crossover operation. In GA, crossover is the second operator which is applied with a pre-defined probability to two selected individuals of a population to generate new offspring of new generation. These offspring inherit some features from parents. Higher fitness chromosome has an opportunity to be selected more than lower one, so good solution always alive to the next generation [18] [2].

The simplest and most popular one [1][2][4][5][6][10] [13][14][16][18][20][22][24] is to choose single point randomly within the chromosome and copy the values of parents 1 and 2 before or after this point to the same locations in the new offspring 1 and 2. Then, the values after or before this point are exchanged by copying them to the new offspring such that genes of parent 1 are copied to offspring 2 and that of 2 are copied to offspring 1. The drawback of this method is that best building blocks can be broken. Also the offspring may have lower performance than parents unless there is restriction on exchanging the genes. The third drawback is that if the cross point happen to be close to one edge of the chromosome then the generated offspring will be very similar to the parent

Another technique used for crossover which overcomes the last drawback of 1-point crossover is known as the two-point crossover [17]. It is similar to the 1-point crossover except that two points are selected randomly as crosspoints and genes between them are exchanged to form the offspring. This technique provides wider diversion from parents than 1-point crossover do, and researchers agree that 2-point crossover is generally better than 1-point crossover [5]. However, if the crosspoints are close to each other then the offspring will not much differ from the parents. This technique is generalized by introducing *n-point* crossover

ISBN: 978-988-19252-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

[11][13][22]. In *n-point* crossover the operation is done by randomly choosing a number of crossover points and applying *n* simple crossover operations on the parents simultaneously. However, adding more crosspoints affects the speed of the crossover process and also disrupts the building blocks.

Other techniques for crossover include *uniform crossover* which is applied by [5][23]. It is implemented in two ways. The first one is to generate a binary mask randomly with the same number of components of the chromosome. Each mask is used to generate a child from a pair of parents. The binary values zero or one in each mask are used to select the value of genes from either the first or the second parent, respectively. The second way of implementing *uniform* crossover is to define a swapping probability *pswap* and perform swapping between parents if the generated random is less than *pswap* for each gene.

Inversion is another technique used for crossover in which the order of genes between 2 randomly chosen positions within the chromosome is reversed [9]. Hence it is applied to a single parent to produce a single offspring.

Similar to inversion is the *reordering* crossover technique but it is applied to two parents to produce two offspring [5]. It is applied to 1-point or 2-points crossover where the order is maintained after each crosspoint. The purpose of *reordering* is to find gene ordering which have better evolutionary potential [5].

In *fusion crossover* [22], only one offspring is generated from two parents where for each gene, the child inherits the value from one or the other of the parents with a probability according to its performance.

III. The Hybrid Crossover Technique

In order to produce high quality offspring, some drawbacks have to be avoided in the proposed crossover technique while others could be overlooked or reduced by combining several techniques together. The main drawbacks that need to be avoided are generating lower performance offspring, breaking building blocks, generating offspring out of search space and low speed of convergence. These drawbacks are to be avoided in the proposed crossover which is called *hybrid crossover*.

A. The Design of Hybrid Crossover

The proposed crossover operator chosen to be implemented here is a combination of reordering crossover [22], fusion crossover [22] and one-point crossover. When genes within a chromosome are ordered based on their fitness value and the order is important, then the crossover applied to such chromosomes is called a reordering crossover. In fact, the order of genes in the proposed crossover is important as it represents the ranked documents that will be displayed to the user. If one offspring is to be produced from the crossover process rather than two, then it is called a fusion crossover. Combining these two techniques together and applying a one-point crossover on them forms the new crossover suggested in this paper.

In the one-point crossover, GA selects one point randomly to perform exchange of genes. A reordering crossover is applied to chromosomes having their genes ordered based on their fitness value from higher to lower. Since genes are in order within the chromosome then a 2-point crossover could not produce better results as the high quality genes are on one edge while the exchange is done for the genes somewhere in the middle. Other techniques of crossover are not applied to the proposed model due to their disadvantages mentioned earlier.

The rationale behind using the ordered crossover technique over other techniques is the need to inherit the good genes and maintain the good building blocks while passing them to the resulting offspring.

In fusion crossover, only one offspring is generated from the two selected parents. In this technique, the offspring inherits the genes from one of the parents with a probability according to its performance. The advantage of this technique is that the good genes of both parents are inherited simultaneously to the offspring, producing high quality offspring and increasing the speed of convergence.

Combining the three techniques of crossover into one process allows fast convergence with high quality offspring. The ordered technique gathers the good genes into one side of the chromosome. Then the one-point crossover copies these gathered genes from the heavy side of both parents to one offspring only. This results in an offspring having the best genes of the parents.

B. The Functionality of Hybrid Crossover

The hybrid crossover operates in the following manner. Suppose there are two parents x and y of length L. These two individuals are selected randomly using binary tournament selection from current population p_i to produce one offspring O of population p_{i+1}. Firstly, the chromosome's genes are ordered based on their fitness value from higher to lower from the previous generation. Then a one-point crossover is applied by choosing crosspoint *cp* randomly over the range [1.. L]. The selected crosspoint divides the chromosomes into two parts. The first O's genes [O_0, .., O_{cp}] are copied from the candidate parent that has the greatest gene's value at position L_0, suppose it is *x* in this example. The remaining genes of O are copied from the second parent starting from the leftmost position until the offspring O is filled up or until it reaches the specified location *cp*. Through the process of copying the remaining genes from the parents, the uniqueness of the copied gene must be considered, i.e., each gene can occur only once in the new offspring O. This is implemented by excluding the genes that already exist in O. When O is not filled up to the specified length, the fitness values of other genes in both parents are compared starting from location *cp*+1. The gene that has a higher fitness value contributes to O. This is done in order to generate offspring with appropriate genes from each parent and to guarantee that the length of O is maintained at L. Figure 1 gives an example of the proposed crossover in which numbers in each chromosome represent the fitness value of the gene at that position. The two candidates *x* and *y* that are shown in Figure 1 -Step A. The crosspoint *cp* is selected randomly to perform a one-point crossover- Step B in Figure 1. In this example it is 3. Because the first gene of *x* has a greater fitness value than the first gene of *y*, *x*'s genes along with the

ISBN: 978-988-19252-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

fitness values are considered as the first three genes of O. To complete the genes of O, the other three genes are copied starting from the leftmost position of *y*. Then a competition between the genes in both *x* and *y* is done to complete the creation of O. Because the gene at position *cp*+1 in y has a greater value than that of *x*'s, then *y*'s genes are copied into O (the right bold set of genes in step C in Figure 1). Once all positions in the offspring are populated with genes, these genes are ordered from higher to lower based on their fitness value (step D in Figure 1). The steps of this hybrid crossover are illustrated in Algorithm 1.

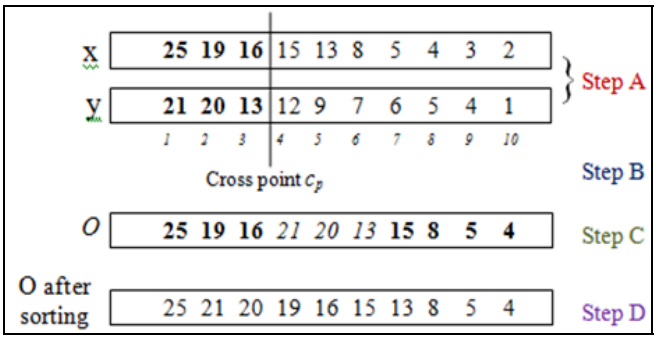

Fig. 1, Illustration of the hybrid crossover process

Prerequisite: Both parents are of same length and the genes in each of them are sorted with respect to their fitness value.
begin
1. Select crosspoint *cp* randomly such that 0 < *cp* < parent length.
2. g_{max}= max gene(f(x_0), f(y_0)) --compare fitness value of first gene in both parents
3. parent 1= chromosome with g_{max}
4. Create offspring such that: *O*
   = g1, g1 ≤ *cp*
   = g2, g2 ≤ c_p, g2 ∉ *O* and length(*O*) ≤ length(parent1)
5. If length (*O*) < length(parent1)
   begin
   $g`_{max}$= max gene(f(x_{cp+1}), f(y_{cp+1}))
   parent 1`= chromosome with $g`_{max}$
   Copy genes from parent 1` to *O* such that genes are unique in *O*
   end;
6. Order genes in *O* in descending order with respect to their fitness value.
end

Algorithm 1, The hybrid crossover operator

IV. Document Set Description

In the GA system, the adoption of effective way to represent documents has greatly influenced the scientists' thought. Actually, the documents that will be evaluated by IR system can be either plain text, semi-structured (i.e., HTML (HyperText Markup Language) documents) or structured. Because most of web-documents are written in HTML (Kim and Zhang, 2003), this format is adopted for implementing our proposed system.

In similar studies, researchers tend to use ready-made data sets which use vector space indexing models such as TREC and ACAM data sets. These sets include documents, vector space index, queries and their results. However, these sets are not suitable for the proposed model because of the indexing model on the one hand, and due to the additional data that need to be included in the index which is not supported by these data sets on the other hand.

The document set or search space used in this work is a set of HTML web documents. This set is the Carnegie Mellon University data set (WebKB). It is a set of HTML documents from the departments of computer science at various universities collected in January 1997 by the World Wide Knowledge Base project of the CMU text learning group. It consists of 8284 documents [21] and used by several researches [8]. This set consists of seven categories, named: course, department, faculty, project, staff, student and others, in additional to another 60 web documents downloaded from the Web by passing different keywords to the Google search engine. Hence, the total number of HTML documents in the set is 8344. Table 1 shows the categories of the document set as well as the number of documents in each category. This document set is expected to be reasonable to analyze the proposed model since this size is in the range of document size used in similar researches. In the literature, the data set used to test most GA-based IR systems is CISI [2][6][18]. This data set consists of 1460 documents and was tested against 76 to 112 queries. Table 1 shows some statistics for the documents and queries used to test the proposed system.

TABLE I. Statistics of the Test Collection Used in the Proposed Model

Parameter Name	Value
Number of documents	8344
Number of queries	100
Number of unique indexed terms	128213
Average number of terms by query	2.69
Average number of relevant documents by query	16.82
Average number of indexed terms by document	410.28

V. Evaluation Measures

The results of the proposed system are evaluated by using precision and recall measures. Precision is defined as the percentage of relevant retrieved documents to the total number of retrieved documents, while recall is defined as the percentage of relevant retrieved documents to the total number of relevant documents.

One of the most popular measures used to evaluate the IR systems is called average precision-recall measure (P@R) where it is used in [2] [6][12][17][18] [19] [24]. It measures the precision at multiples of 10% of the total relevant retrieved documents for the given query. In other words, if the query has 100 relevant documents, then this measure will evaluate the precision when retrieving 10, 20, 30,.., 100% of the relevant documents. Therefore, this measure evaluates the system in terms of percentage of the total relevant documents.

In addition to the average precision-recall measure, two common measures are used to evaluate such systems. These measures are: Precision at Rank N (P@N) and Recall at Rank N (R@N), where N is multiples of 10 [3][12]. Rank N here means the top N ranked documents of the retrieved

ISBN: 978-988-19252-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

documents. In this method, the retrieved documents are ranked in descending order based on the fitness value and the average of precision and recall are calculated. Therefore, this measure evaluates the system based on the percentage of the total retrieved documents.

When the maximum value of N is 100, this measure is called 11-point average precision [3][15] and it is widely used to evaluate IR models, since it measures the performance at the points 0, 10, 20, 30 up to 100 top ranked retrieved documents, where point 0 means the first retrieved document or the top ranked document.

VI. Experiments

In this experiment, three measures are applied to study the performance of the *hybrid crossover* technique which performs one-point crossover on ordered parents to produce one ordered offspring. These measures are P@N, R@N and P@R. These measures will be applied to compare the hybrid crossover technique with other three well known crossover techniques. These techniques are: 2-point crossover (2-point CO), non-ordered crossover (Non-Order-CO) and one-point crossover producing 2 offspring (2-Offspring).

A. Precision @ N Measure

The first measure to be analyzed is the P@N measure using the above four mentioned crossover techniques. Figure 2 shows the P@N retrieved documents. It is shown that the proposed system using hybrid crossover has much better performances than the other crossover techniques. In details, the hybrid crossover achieves 0.86 at top 10 retrieved documents while the non-ordered achieved 0.58, 2-offspring CO achieved 0.48 and 2-point crossover achieved only 0.34. The average 11-point precision of the hybrid crossover is 0.44 which is higher than the non-ordered crossover by 61.36%, higher than 2-point crossover by 236.07% and higher than 2-offspring by 237.97%. The non-ordered crossover technique starts with 0.58 at P@10 and ends with 0.13 at P@100. That means the hybrid crossover technique is enhanced from 48.12% to 70.62% compared to the non-ordered crossover.

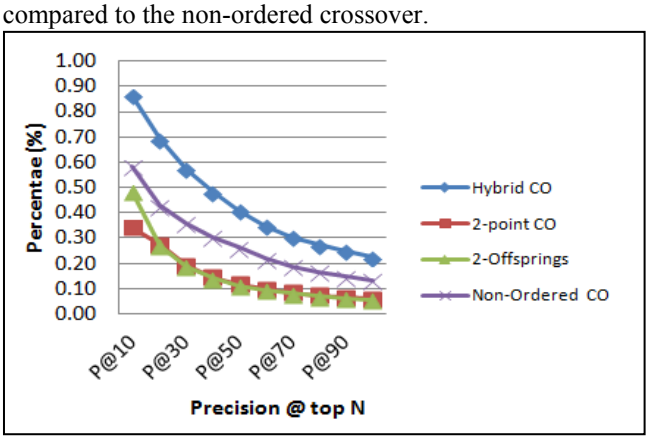

Fig. 2, Precision @ top N retrieved documents for different crossover techniques

B. Recall @ N Measure

The second measure to be considered in evaluating these techniques is the R@N which is depicted in Figure 3. The hybrid crossover starts from 63% till reaches 85% at R@60.

That means this technique is capable of retrieving 85% of total relevant documents at top 60 retrieved documents. However, 2-point crossover technique starts by retrieving 31% of relevant documents at top 10, and as a whole it retrieves only 48% at top 100 retrieved documents. That implies hybrid crossover achieves enhancement of 104% at R@10 and drops to enhancement of 79.16% at R@100.

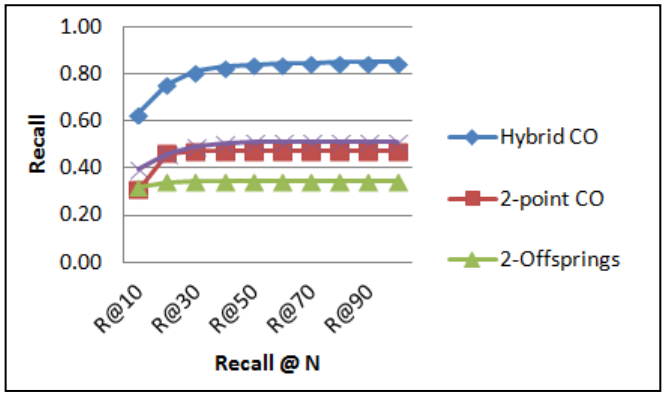

Fig. 3, Recall@N improvement by hybrid crossover

Looking at the performance of the 2-offspring technique, it is shown that it retrieves 32% of the relevant documents at top 10 and increases to 35% at top 100 retrieved documents. These results are of low performance when compared to that of the hybrid crossover which retrieves 63% at top 10 and retrieves 86% at top 100 retrieved documents. In another words, the hybrid crossover enhanced the performance from 98.42% at top 10 retrieved documents to 145.7% at top 100 retrieved documents.

When comparing the non-ordered crossover with the hybrid crossover technique, it is noticed that the non-ordered crossover performance ranges between 39% at R@10 and 51% at R@100. This means that this technique is lagging behind the hybrid crossover technique by 60.33% to 68.11%.

When comparing the non-ordered crossover with the hybrid crossover technique, it is noticed that the non-ordered crossover performance ranges between 39% at R@10 and 51% at R@100. This means that this technique is lagging behind the hybrid crossover technique by 60.33% to 68.11%.

C. Precision @ Recall Measure

The third measure is the P@R which evaluates the precision percentage when retrieving multiples of 10% of the relevant documents. In another words, this measure evaluates the purity of the results from the irrelevant documents.

From Figure 4, one can deduce the high difference in performance between the hybrid crossover technique from one side and the other techniques from other side, where hybrid crossover reaches its maximum precision value of 1when retrieving 10% (P@R10) of relevant documents at the time where 2-offspring crossover technique reaches its maximum of 0.79 at the same point. In average, the hybrid crossover technique has enhanced the P@R measure by 114.69% over the 2-offspring crossover as illustrated in Table 5.

The 2-point crossover technique is the second best

ISBN: 978-988-19252-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

technique analyzed here. It has 22% of irrelevant documents when retrieving 10% of relevant documents. And this percentage rises to 69% when retrieving all relevant documents. These scores show that the hybrid crossover managed to achieve enhancement of 130% in average over all 10-points shown in Figure 4 for this measure.

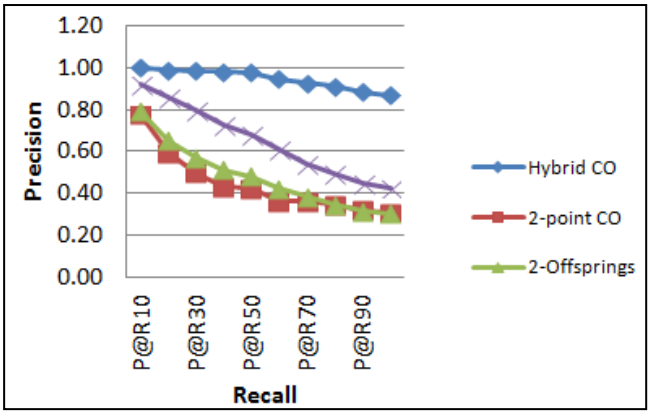

Fig. 4, Precision @ Recall improvement by hybrid crossover

D. Assessment of the Results

The explanation of the results presented in the previous sections is provided here starting by the 2-point crossover. Since the genes within each parent are ordered according to the fitness value, then expecting that in 2-point crossover, the offspring will not much differ from parents as the genes at each edge which form the extremes of best and worst documents are copied as they are to the offspring while the middle genes which have medium performance are exchanged causing the offspring to differ slightly from parents. Hence its performance is low compared to the hybrid crossover.

Another alternative crossover technique is the 1-point crossover applied on non-ordered chromosomes to produce one offspring. What differentiate this technique from the hybrid crossover technique is that the genes within the chromosome are not ordered based on their fitness value. Thus, good genes (genes that have high fitness value) are scattered among the chromosome resulting in a chromosome having mixture of good and bad genes distributed arbitrary within the chromosome. Applying one-point crossover on such chromosome results in swapping these mixed genes from one side of the cross point to another side without any noticeable improvement.

Finally, in the 2-offspring crossover technique, the good genes are concentrated at the left side of the chromosome. When creating the offspring, these good genes are swapped between the offspring, resulting offspring of similar or close performance to that of the parents. Hence the overall improvement across the generations is low causing low performance

VII. Conclusion

This paper proposed a new hybrid crossover technique as an operator of the genetic algorithm. It is constructed by applying 1-point crossover to the ordered chromosome to produce one offspring which combines the best genes of both parents. This technique is applied as part of the genetic algorithm to retrieve HTML documents based on user query. Its performance is compared with 2-point crossover, non-ordered crossover and 1-point crossover that produces two offspring. This technique achieved highest score among these three techniques in terms of recall, precision and precision-recall measures. To generalize the results and further demonstrate its efficiency in the IR domain, it needs to be compared with other crossover techniques such as the uniform crossover, and need to be applied to larger document set. This work is applied on chromosome with fixed length and there is a need to examine the performance if the length of the chromosome is changeable

References

[1] Al-Hajri, M.T.; Abido, M.A., (2009), Assessment of Genetic Algorithm selection, crossover and mutation techniques in reactive power optimization, IEEE Congress on Evolutionary Computation, 2009. CEC '09. Publication Year: 2009 , pp. 1005 - 1011

[2] Aly, A. (2007). Applying genetic algorithm in query improvement problem. Information Technologies and Knowledge , vol.1, pp. 309-316.

[3] Alzahrani, S.M.; Salim, N., (2009), On the use of fuzzy information retrieval for gauging similarity of Arabic documents, Applications of Digital Information and Web Technologies, ICADIWT '09. Second International Conference on the Digital Object, pp.: 539 – 544. IEEE Conference Publications.

[4] Asllani, A. and Lari, A. (2007). Using genetic algorithm for dynamic and multiple criteria web-site optimizations. European Journal of Operational Research, vol. 176, no.3, pp. 1767-1777.

[5] Beasley, D., Bull, D. R. and Martin, R. R. (1993). An overview of genetic algorithms: part 2, Research Topics. *University Computing*, vol. 15, no. 4, pp. 170-181.

[6] Billhardt, H., Borrajo, D. and Maojo, V. (2002). Using genetic algorithms to find suboptimal retrieval expert combinations. In Proceedings of SAC, pp. 657-662.

[7] Desjardins, G., Godin, R., and Proulx, R. A. (2005). Genetic algorithm for text mining. Proceedings of the 6th international conference on data mining, text mining and their business applications, vol. 35, pp. 133-142.

[8] Dong, H., Hussain, F. K., and Chang, E. (2008). A survey in traditional information retrieval models. Second IEEE International conference on digital ecosystems and technologies, pp. 397 - 402.

[9] Goldberg, D. E. (1989). Genetic algorithms in search, optimization, and machine learning. Addison-Wesley.

[10] Húsek, D., Snášel, V., Owais, J., and Krömer, P. (2005). Using genetic algorithms for boolean queries optimization. Proceeding of the Ninth IASTED International Conference internet and multimedia ststems and applications, pp. 178-184. Honolulu, Hawaii, USA.

[11] Kazarlis, S. A., Papadakis, S. E., and Theocharis, J. B. (2001). Microgenetic algorithms as generalized hill-climbing operators for GA optimization. IEEE Transaction on Evolutionary Compution,vol 5, pp. 204-217.

[12] Kim, S., and Zhang, B-T. (2003). Genetic mining of html structures for effective web-document retrieval. Applied Intelligence, vol.18, no.3, pp.243-256.

[13] Klabbankoh, B., and Pinngern, O. (2008). Applied Genetic Algorithms In Information Retrieval. Retrieved Aug 22, 2009, from http://www.ils.unc.edu/~losee/gene1.pdf

[14] Lopez-Pujalte, C., Guerrero-Bote, V. P., and de Moya-Anegon, F. (2003). Genetic algorithms in relevance feedback: a second test and new contributions. Information Processing and Management, vol. 39, pp. 669–687.

[15] Manning, C. D., Raghavan, P., and Schütze, H. (2009). An introduction to information retrieval. Cambridge, England: Cambridge University Press

[16] Marghny, M. H., and Ali, A. F. (2005). Web mining based on genetic algorithm. AIML 05 Conference. Cicc, Cairo, Egypt.

ISBN: 978-988-19252-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

[17] Pathak, P., Gordon, M., and Fan, W. (2000). Effective information retrieval using genetic algorithms based matching functions adaption. 33rd hawaii international conference on science (HICS). Hawaii, USA.

[18] Radwan, A. A., abdel Latef, B. A., Ali, A. A., and Sadeq, O. A. (2006). Using genetic algorithm to improve information retrieval systems. proceedings of world academy of science, engineering and technology, vol. 17, pp. 6-12.

[19] Saini, M. Sharma, D. Gupta, P.K . (2011), Enhancing information retrieval efficiency using semantic-based-combined-similarity-measure. International Conference on Image Information Processing (ICIIP), pp. 1 - 4. IEEE Conference Publications

[20] Song, W., and Park, S. C. (2009). Genetic algorithm for text clustering based on latent semantic indexing. Computers and Mathematics with Applications , vol. 57, no.11, pp. 1901-1907.

[21] The 4 Universities Data Set. (1998). [online]. Available at:
http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/ [Accessed 12/11/2009]

[22] Vrajitoru, D. (2000). Large population or many generations for genetic algorithms ? implications in information retrieval. In F. Crestani, and G. Pasi (Ed.), Soft Computing in Information Retrieval. Techniques and Applications (pp. 199-222). Physica-Verlag, Heidelberg.

[23] Xu, Y., Deli, Y. and Yu, L. (2009), Efficient annealing -inspired genetic algorithm for information retrieval from web-document June 2009, **GEC '09:** Proceedings of the first ACM/SIGEVO Summit on Genetic and Evolutionary Computation, **Publisher:** ACM.

[24] Yeh, J.-Y., Lin, J.-Y., Ke, H.-R., and Yang, W.-P. (2007). Learning to rank for information retrieval using genetic programming. In Proceedings of ACM SIGIR 2007 Workshop on Learning to Rank for Information Retrieval (LR4IR '07), pp. 41-48. Amsterdam, Netherlands.

ISBN: 978-988-19252-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)