EFFECTIVE INFORMATION RETRIEVAL METHOD BASED ON MATCHING ADAPTIVE GENETIC ALGORITHM

[1]ESSAM HANANDEH, [2]KHALED MAABREH

[1]Department of Computer Information Systems, Faculty of IT, Zarqa University, Zarqa, Jordan

[2]Department of Computer Information Systems, Faculty of IT, Zarqa University, Zarqa, Jordan

E-mail: [1]hanandeh@zu.edu.jo, [2]kmaabreh@zu.edu.jo

ABSTRACT

Information Retrieval (IR) System is very complex in nature due to the complex interactions between documents and queries, which means that the matching of document representations and query representations is not straightforward. The Genetic Algorithm (GA) is widely used in IR systems to improve the effectiveness such systems. This study uses the Vector Space Model (VSM) and the Extended Boolean Model (EBM) to compute the similarities between queries and documents. Two fitness functions are proposed in this paper: One as fitness function and the other as adaptive mutation. Then comparing each of these functions with a number of ratio mutations that have been introduced to get better results. The experimental results reveal that the proposed cosine function outperformed other fitness models.

Keywords: *Information retrieval system, Genetic algorithm, Adaptive mutation, Vector Space Model, Extended Boolean Model,* Mutation

1. INTRODUCTION

In information retrieval systems, there are many fundamental processes that are used to access the information that meets user needs by finding similarities between the query and the existing documents. This study presents basic processes for matching the keywords with the existing documents by using both VSM and EBM [3, 11]. In order to determine the best method and use its results as an input to GA, each query is considered as a chromosome in GA, which used simulations in the problem-solving process. The simulation techniques used a random search to find the best solution, or the best among all solutions, for the available data [2, 12, 19]. The keywords are extracted from relevant documents; those relevant keywords are then appointed and converted to a binary system called chromosomes. In this case, the use of GA will help the user to get the best solution [3, 21].

In this paper, two fitness functions are proposed: One as fitness function and the other as adaptive mutation. To achieve an exhaustive improvement of IR systems using the genetic algorithm, the results of those functions will be compared with GA based on Cosine Fitness function and Jaccard Fitness function in VSM, and will be also compared with the Cosine Fitness function and Jaccard Fitness function in EBM. We try to enhance the IR system performance by using the adaptive genetic algorithm, which can improve the quality of the query, and may get the most advanced queries that fit the needs of the researchers. This includes investigating and evaluating various fitness functions as well as reducing the search space, which leads to saving time and reducing the number of iterations required to generate the most optimized query. Furthermore, the study determines the best technique to modify the query in the IR system. This research has introduced a new perspective to the area of matching function adaptation in IR, whereas the previous studies have focused primarily on document and query adaptation. We have shown here that the genetic adaptation of matching functions can lead to improved retrieval performance. However, more work needs to be done in this area.

2. GENETIC ALGORITHM

GA is one of the main techniques used in a random search to look for the best solution, or the best among a group of solutions, in the available data [16, 20]. In a genetic algorithm, a population of candidate solutions to an optimized problem evolves toward better solutions. Each candidate solution has a set of properties (its chromosomes) that can be mutated and altered. Traditionally, solutions are represented in binary form as strings of zeroes and ones, but other encoding styles are also possible [1, 4].

GA operates on the encoded representation of the solutions, equivalent to those chromosomes of individuals in nature. It is assumed that a potential solution to a problem may be encoded as a chromosome [7, 14].

2.1 Genetic Algorithm Approach

The implementation steps of the GA approach are:

1- **Representation of the chromosomes:** Chromosomes are expressed as a stream of binary representation. They represent a group of genes that depend on the degree of similarity between the account inquiry and the declaration. In this study, 30 documents were selected and transformed to a binary system. Each chromosome is represented by the keyword entered to inquire. If the word is found in the document, then it takes one (1), else it takes zero (0) [18].

2- **Fitness Function:** This is a measure used to compute performance or reward functions. It assesses the solution and determines whether it was perfect or not. The genetic algorithm uses two fitness functions, Cosine and Jaccard's with modification, to find the similarity between chromosomes. Two fitness functions are used to get better results during measuring. Then, the equation will be changed to fitness and will seek to get the best results from the update that took place during the equation.

3- **Selection Method:** The genetic algorithm is used for selecting a mechanism against a random sample of the application of operations. This mechanism consists of the construction based on selecting a random chromosome. After the operation of genetic algorithm made swapping. It adds the better one for each pair of chromosomes to the population. This means that the best chromosome is added to achieve a greater proportion [13].

4- **Operators:**

a. **Crossover:** This is the genetic operator that mixes two chromosomes together to form a new offspring. The mixing is done based on adaptive probability using crossover to get better value [6, 10].

b. **Mutation:** The second operator used in our GA system includes a mutation to modify the values of the gene under short changes; some values of the chromosomes will give a different breed. The goal of the boom is to restore the lost and explore a variety of data. Then, the adaptive ratio mutation will be used based on applying one point randomly [9, 10].

3. USING GA WITH INFORMATION RETRIEVAL SYSTEMS

3.1 Building Genetic Algorithm:

Representation of the chromosome: The genetic algorithm uses the initial population that expresses the chromosome. Each chromosome consists of several genes. Each chromosome is calculated using a number of query terms that we consider as 1, if the term exists in the document, and 0, if it does not exist. The system snapped the first 30 documents retrieved by the system and used these documents in the initial population in the adaptive genetic algorithm (AGA). Finally, each chromosome in the population will be evaluated by using adaptive and modified fitness functions Cosine and Jaccard. The proposed fitness function after modification is:

$$F\,(cosine) = \left(\frac{\sum_{i=1}^{t} w_{i,j}\, w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^{2} \sum_{i=1}^{i} w_{i,q}^{2}}} \right) + \left(\frac{\sum_{i=1}^{t} w_i q_i)}{2} \right) \text{........ } \mathbf{(1)}$$

Where $w_{i,\,j}$= weight of term i in document j.,
$w_{i.\,q}$= weight of term i in query q.

$$F\,(Jaccard's) = \left(\frac{\sum_{i=1}^{t} w_{i,j}\, w_{i,q}}{\sum_{i=1}^{t} w_{i,j}^{2} + \sum_{i=1}^{t} w_{i,q}^{2} - \sum_{i=1}^{t} w_{i,j}\, w_{i,q}} \right) + \left(\frac{\sum_{i=1}^{t} w_i q_i)}{2} \right) \text{.... } \mathbf{(2)}$$

Where $w_{i,\,j}$= weight of term i in document j.,
$w_{i.\,q}$= weight of term i in query q.

Selection method: The selection process consists of selecting two parent chromosomes on the basis of random fitness function to achieve a better owner of fitness and the greatest opportunity to choose.

3.2 Genetic Operations:

Crossover: The process of mixing two chromosomes to get two new chromosomes as offspring that can be used in the genetic algorithm, where the crossover is used to get a better chromosome. Multiple trials mean that more accurate results are obtained through AGA [8]. To find the value of the crossover, equation (3) [8] will be used.

$$Pc=\begin{cases} \dfrac{K_1(f_{\max}-f')}{f_{\max}-f_{avg}}, & f \geq f_{avg} \\ K_2, & f < f_{avg} \end{cases} \qquad \ldots\ldots\ldots(3)$$

Where K_1 = 0.5, K_2 = 0.9 is the crossover probability.

f_{max} = maximum fitness function in the chromosome that is chosen.

f' = fitness function in the chromosome.

f_{avg} = average fitness function for all chromosome fitness.

Mutation: The process of changing one bit is based on random selection to get a better chromosome. Multiple iterations mean more accurate results in the work of AGA [4]. A random one-point mutation operator in the range of probability will be used, but (0.001, 0.2) is usually used with adaptive based on an equation (3). Then, a random number will be selected and compared with the possibility of mutation. If the number is smaller than or equal to the probability of the mutation is then applied, but nothing will occur if the condition is not satisfied.

The equation to find the value of the mutation operator [8]:

$$Pm=\begin{cases} \dfrac{K_3(f_{\max}-f')}{f_{\max}-f_{avg}}, & f \geq f_{avg} \\ K_4, & f < f_{avg} \end{cases} \qquad ----(4)$$

Where K_3 = 0.2, K_4 = 0.001 are mutation probabilities.

f_{max} = maximum fitness function in the chromosome that is chosen.

f' = fitness function in the chromosome.

f_{avg} = average fitness function for all chromosome fitness.

Evaluation: The algorithm is stopped after attempting to find the best number of generations to get the optimal solution. Recall and precision are used to evaluate these documents to compare the use of traditional Cosine and Jaccard's efficiency of fitness functions and the use of the proposed adaptive fitness function.

4. PROPOSED ALGORITHM

This section presents the essential steps of the proposed algorithm

Step 1: Tokenizing:
To extract tokens/words from documents.

Step 2: Removing stop words:
Remove any word that has no such meaning as “a”, “the”, “to”, “be”, etc.

Step 3: Stemming:
To determine the root of English words using Porter stemmer.

Step 4: Inverted index:
Represent frequencies of the terms and where they can be found.

Step 5: Assigning term weights:
A similarity weight is assigned to each term extracted from the documents.

Step 6: Using two models, VSM and EBM:
To describe document contents in a corpus and compute the similarity between query and documents.

Step 7: Using the cosine measure:
To compute similarity between documents and queries using a cosine measure.

Step 8: Initializing population:
The system snapped the first30 documents retrieved by the system; these documents will be the initial population.

Step 9: Representation of the chromosome:
Represent each individual population in the binary presentation.

Step 10: Evaluate fitness:
Use a fitness function to get better population by using adaptive cosine fitness and adaptive Jaccard's fitness.

Step 11: Selection method:
Choose two chromosomes based on random selection.

Step 12: Crossover:
Achieve a new, better solution based on two points after the second gene.

Step 13: Mutation operator:
Achieve a new, better solution based on one point randomly.

Step 14: Evaluation:

Evaluates two offspring with two chromosomes in order to choose two chromosomes among them.

5. LITERATURE REVIEW

Korejo and Khuhro (2013) [2] studied adaptive mutation and proposed four operators in the genetic algorithm to determine the operator mutation in spite of the difficulty of the matter in the application process. They proposed a solution of adapting the mutation percentage of mutation by choosing each operator mutation, according to the behavior of the initial population in each generation. Their study has shown that the work of adaptation mutation gave the best result.

Ammar (2012) [5] proposed a research method based on the genetic algorithm to improve the IR system for websites and to apply IR using a genetic algorithm to divide the work into two units—the document indexing unit and the genetic algorithm unit—by the use of crossover, mutation operator, and specialized fitness function. The proposed approach obtained an improvement rate of up to 90%.

AL-Mashakbeh (2008) [17] uses different strategies of genetic algorithms. Each strategy in his Ph.D. thesis is based on the difference between the GA operators (crossover and mutation). His study is applied with different mutation strategies and different fitness functions (recall, precision) on a Fuzzy Set. He noticed that GA with a point mutation gave a higher improvement to the effectiveness of the IR system than traditional IR methods. Maitah et al. (2013) [3] study investigated the use of the adaptive genetic algorithm (AGA) under the vector space model, Extended Boolean model, and Language model in (IR).The algorithm used crossover and mutation operators with variable probability; whereas a traditional genetic algorithm (GA) uses fixed values of those and remains unchanged during execution. GA is developed to support the adaptive adjustment of mutation and crossover probability, which allows for faster attainment of better solutions. The Genetic Algorithm gives the highest improvement over the vector space model with Cosine as fitness under the traditional genetic algorithm with 55.1%.

A framework for generating and refining the matching functions used for document ranking is proposed by Weiguo and Gordon (2009) [15]. The framework combines the merits of genetic programming and relevance feedback techniques for document ranking. Their approach overcomes the drawbacks of traditional ranking algorithms and provides some hints for information retrieval professionals.

6. EXPERIMENTAL RESULTS

This section presents the results of the series of experiments conducted to evaluate the effectiveness of the proposed algorithm. These experiments include the use of 10 queries with 8 states.

Table 1 presents the results of the conducted experiments to evaluate AGA with the VSM model. Table 2 exhibits the results of testing the effectiveness of the EBM model to compute the similarities between each query and documents.

Table 3 shows the average value of the precision for both models (VSM, EBM). Finally, Table 4 presents the comparison between proposed Cosine and proposed Jacard models, and shows the degree of improvement for each IR model (VSM, EBM) with each of the fitness functions.

Table 1: Average value of precision for all queries using VSM

Recall	**Average Precision of Cosine (%)**	**Average Precision of Proposed Cosine (%)**	**Average Precision of Jaccard's (%)**	**Average Precision of Proposed Jaccard's (%)**
0.1	85	92	80	87
0.2	76	85	71	76
0.3	72	80	62	70
0.4	65	68	57	65
0.5	51	55	42	47
0.6	39	45	32	35
0.7	33	38	30	31
0.8	23	28	21	25
0.9	20	23	19	20
Average	51.5	57.1	46	50.6

Table 2: Average Value of Precision for All Queries Using EBM

Recall	Average Precision Cosine (%)	Average Precision Proposed Cosine (%)	Average Precision Jaccard's (%)	Average Precision Proposed Jaccard's (%)
0.1	79	89	82	89
0.2	70	75	73	77
0.3	61	70	64	75
0.4	59	65	59	64
0.5	44	49	47	53
0.6	33	36	36	41
0.7	30	32	31	36
0.8	21	25	21	23
0.9	19	21	19	20
Average	46.2	51.3	48	52.8

Table 3:Average Value Of Precision for VSM And EBM

Option	Cosine (VSM)	Proposed cosine	Cosine (EBM)	Proposed Cosine	Jaccard 's (VSM)	Proposed Jaccard's	Jaccard's (EBM)	Proposed Jaccard's
Average	51.5	57.1	46.2	51.3	46	50.6	48	52.8

Table 4: Comparison between Proposed Cosine And Proposed Jaccard's Models

	Average Improvement Proposed cosine (%)	Average Improvement Proposed Jaccard's (%)
VSM	5.6	4.6
EBM	5.1	4.8

The improvement degree with VSM and EBM for the proposed Cosine function is greater than the degree for the proposed Jaccard's, because AGA gave more accuracy in the working system as shown in Figure1 and Figure2, respectively. Table 4 shows the results obtained by using VSM and EBM in proposed Cosine and Jaccard's models. The results indicate that VSM with the proposed Cosine have the best improvement by using an adaptive genetic algorithm. The data on the improvement of average precision in all cases using an information retrieval model (VSM, EBM) is presented in Table 3.

7. CONCLUSION

This paper presents a proposed approach that uses VSM and EBM to compute the similarity between queries and documents in order to increase the IR efficiency within the complicated interaction between documents and the search query. To achieve better results, two fitness functions are modified and used: Adaptive Fitness and Adaptive Mutation operators. The model runs against a combination of fitness functions (Cosine, proposed cosine, Jaccard's, and proposed Jaccard's) on VSM and EBM. With VSM, a comparison of different AGA strategies has been made by evaluating the average recall and the average precision formulas. VSM with proposed Cosine as fitness represents the best strategy over VSM with Cosine as fitness, while the VSM with a proposed Jaccard's as fitness represents the best strategy over VSM with Jaccard's as fitness. In EBM, the researcher compares different AGA strategies by evaluating the use of average recall and average precision formulas. EBM with proposed Cosine as fitness represents the best strategy over EBM with Cosine as fitness, while EBM with a proposed Jaccard's as fitness represents the best strategy over EBM with Jaccard's as fitness. The researcher compares different AGA strategies by calculating an evaluation using the average recall formula. VSM with proposed Cosine as fitness represents the best strategy over VSM with Jaccard's as fitness, the EBM with Cosine as fitness, the proposed Cosine as fitness, with Jaccard's as fitness, and with a proposed Jaccard as fitness. The best result which is 5.6% is obtained when using the proposed Cosine Fitness function. The proposed algorithm is also tested on the Cranfield document collection (Which is a widely used database by the IR systems) and it's worked well and the result look satisfying.

We have included only four matching functions in these experiments, and we need to include more matching functions to enrich the results in future research. It is also necessary to test the algorithm on different document collections to see how it performs with scaling in terms of both the size of the database and the available features

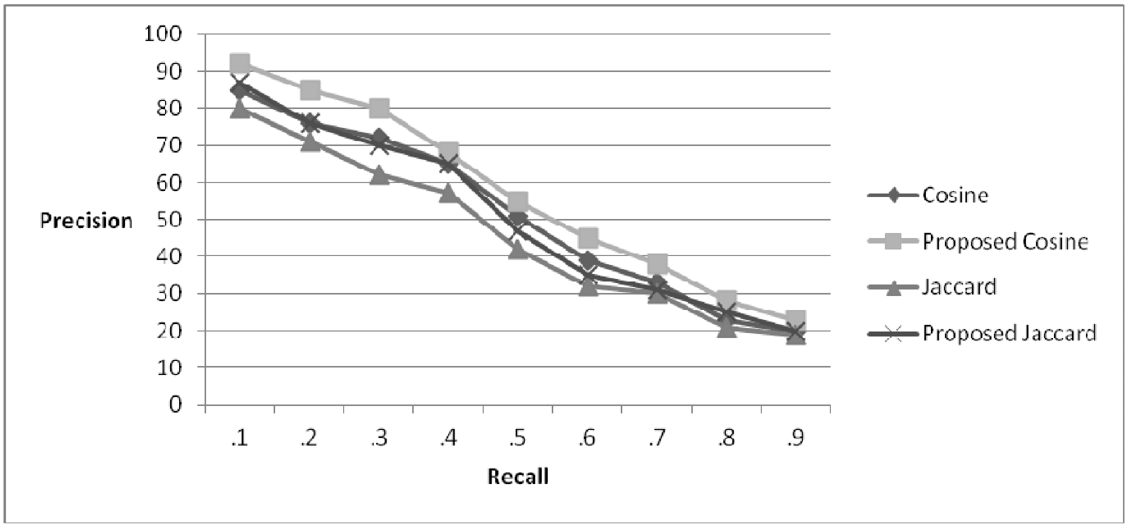

Figure 1: (*Average Precision for VSM*)

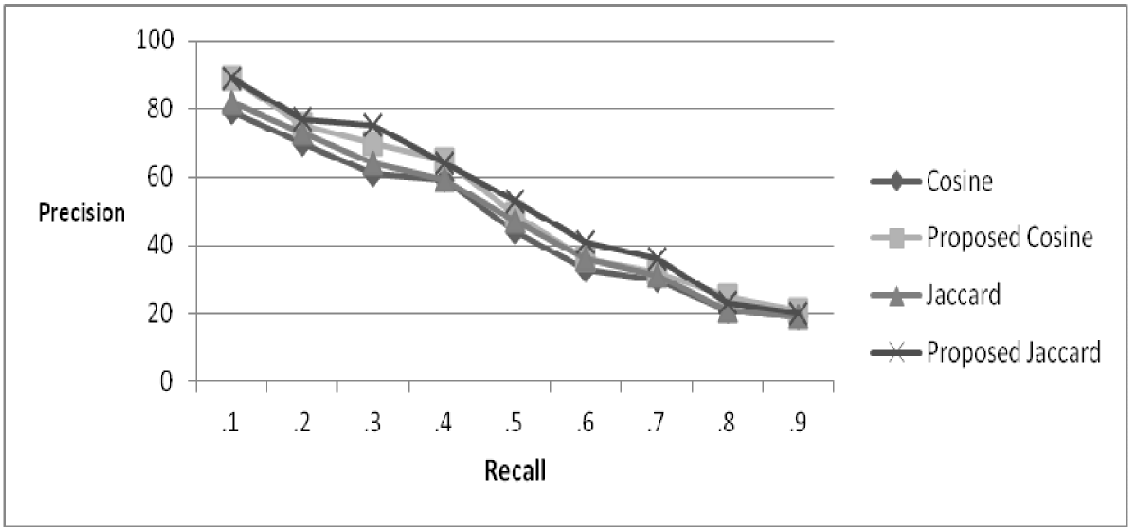

Figure 2: (*Average Precision for EBM*)

REFRENCES

[1] *A. Kalayanasaravan, M. Thangamani, and E. Venkatesh ,* "Document Retrieval System Using Genetic Algorithm", *Kongu Engineering College, Perundurai,* Vol. 2, No.10, 2013, pp. 943-946.

[2] *I.A. Korejo, Z. U. Khuhro, F. A. Jokhio, and F. N. Memon*, "Genetic Algorithm Using an Adaptive Mutation Operator for Numerical Optimization Functions", *Sindh University Research Journal,* Vol. 45, No. 1, 2013, pp. 41- 48.

[3] W. Maitah , M. Al-Rababaa, and G. Kannan, "Improving the Effectiveness of Information Rereieval System Using Adaptive Genetic Algorithm", *International Journal of Computer Science & Information Technology (IJCSIT)*, Vol. 5, No. 5, 2013, pp. 91-105.

[4] Taisir E., "Mutative Genetic Algorithms", *Journal of Computations & Modeling*, Vol. 3, No. 2, 2013, pp. 111-124.

[5] A.S. Al-Dallal, "Enhancing Recall and Precision of Web Search Using Genetic Algorithm", *Diss. Brunel University, School of Information Systems, Computing and Mathematics*, August 2012, Founded on: http://bura.brunel.ac.uk/handle/2438/7379.

[6] G. Desjardins, R. Godin, and R. Proulx, "Performance of Information Retrieval Models Using Term Co-occurrences", *WIT Transactions on Information and Communication Technologies*, 2012.

[7] F. Alabsi, R. Naoum, "Comparison of Selection Methods and Crossover Operations using Steady State Genetic Based Intrusion Detection System", *Journal of Emerging Trends in Computing and Information Sciences*, Vol. 3, No.7, 2012, pp. 1053-1058.

[8] P. Bhatnagar, N. Pareek, "A Combined Matching Function based Evolutionary Approach for development of Adaptive Information Retrieval System", *International Journal of Emerging Technology and Advanced Engineering India*, Vol. 2, No. 6, 2012, pp. 249-256.

[9] M.O. Nassar., F.F. AL-Mashakbeh, E.F. AL-Mashakbeh, "Improving the User Query for the Boolean Model Using Genetic Algorithms" IJCSI *International Journal of Computer Science Issues*, Vol. 8, No. 1, 2011, pp. 66-70.

[10] A. Li, "The Operator of Genetic Algorithms to Improve its Properties", *Modern Applied Science*, Vol. 4, No. 3, 2010, pp. 60-62.

[11] I. Korejo, S. Yang, "A Comparative Study of Adaptive Mutation Operators for Genetic Algorithms", *The VIII Metaheuristics International Conference*, Germany, July 13–16, 2009, pp. 1-10.

[12] H. Cheng, Y. Zhang, and F. Li, "Improved Genetic Programming algorithm", *International Asia Symposium on Intelligent Interaction and Affective Computing*, IEEE, 2009, pp. 168-177.

[13] N. Mezyan, "Web mining based on island genetic algorithm", *Master Thesis, Al Albayt University*, 2009.

[14] P. Simon, p. Sathya, " Genetic Algorithm for Information Retrieval", *International Conference on Intelligent Agent & Multi-Agent Systems*, Chennai, July 22-24, 2009, PP.1-6.

[15] W. Fan, M. Gordan, and P. Pathak, "Automatic Generation of a Matching Function by Genetic Programming for Effective Information Retrieval", *Americas Conference on Information Systems,* 2009, pp. 49-51.

[16] A. A. Radwan, B. A. Abdel Latef, A.M. Ali, and O.A. Sadek, "Using Genetic Algorithm to Improve Information Retrieval Systems", *World Academy of Science, Engineering and Technology*, 2008, pp. 1021-1027.

[17]F.F. AL-Mashakbeh, "Evaluate the Effectiveness of Genetic Algorithm in Information Retrieval Based on Arabic Documents", *PhD Thesis, Faculty of Information System and Technology, The Arab Academy for Banking and Financial Science*, Amman, Jordan, 2008.

[18] A. Ali, "Applying Genetic Algorithm in Query Improvement Problem", *International Journal of Information Technologies and Knowledge*, Vol. 1, No. 1, 2007, pp. 309-316.

[19] S. Uyar, S. Sariel, and G. Eryigit, "A Gene Based Adaptive Mutation Strategy for Genetic Algorithms", *Gentic Evolutionary Computation Conference, Seattle, WA, USA*, June, 2004, pp.26-30.

[20] N. Oren, "Reexamining tf.idf Based Information Retrieval with Genetic Programming", *Conference of the South African Institute of Computer Scientists and Information*, 2002, pp.224-234.

[21]Information Retrieval: http://www.dsoergel.com/NewPublications/HCI EncyclopediaIRShortEForDS, pdf, pp 1-11. Accessed on 11, May, 2015.