ISSN:0975-9646

Anubha Jain et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (4) , 2014, 5921-5927

Relevance of Genetic Algorithm Strategies in Query Optimization in Information Retrieval

Anubha Jain[#1], Swati V. Chande[#2], Preeti Tiwari[#3]

[#1] *Department of CS & IT, The IIS University, Jaipur, India*
[#2,, # 3] *Department of CS, International School of Informatics & Management, Jaipur, India*

***Abstract* -The augmentation of digital information on the Web has proliferated informational needs and expectations of the seekers, resulting in insistent need of more advanced search tools, that are able to respond to the informational requirements within an organization. The user may formulate a search query in a way that can obscure the useful documents to be retrieved. The objective of query optimization is to transform the query into an effective form to improve the quality of recovered information and to reduce the computational burden in processing document text at query time. Genetic algorithms are efficient and robust methods, employed widely in optimization of a variety of search problems, motivated by Darwin's principles of natural selection and survival of the fittest. This paper reviews relevance of genetic algorithms to improve upon the user queries in the field of Information Retrieval.**

***Keywords:* Query Optimization, Information Retrieval, Genetic Algorithm, Genetic Operators, Ranking.**

I. INTRODUCTION

Information is a critical resource for every organization and influences the process of decision making strongly. The advancement in Web technologies has prominently transformed the ways of storing and accessing data, as the resources on the Web are heterogeneous, semi-structured and distributed. Timely availability of relevant information is imperative for effective execution of managerial functions such as organizing, leading, planning and control.

Information retrieval (IR) is a scientific discipline for addressing the problem of finding relevant information in document collections coming from various sources, e.g., enterprises, the Web, etc. The goal of IR systems is to aid the user in storing and organizing this information and retrieving the most similar documents that are likely to satisfy information needs of the user. To resolve this problem, many research communities have implemented diverse techniques such as full text, inverted index, keyword querying, Boolean querying, probabilistic retrieval, neural network, genetic algorithm and machine learning [2].

Heuristic algorithms that obtain approximate solutions with acceptable time and space complexity play indispensable role in query optimization in search data sets. Different heuristic algorithms like evolutionary based optimization algorithms, Swarm-based algorithms and Fuzzy Inference based models have been used for optimizing the query in information retrieval from time to time, by different researchers [1] [5] [8] [10] [14] [28] [30]. In [13], a comparative study of the use of heuristic algorithms for optimization of queries in Information Retrieval has been done. This study attempts to gain a comprehensive view of improvement in Information Retrieval process using heuristic techniques.

Genetic Algorithm (GA) is a probabilistic algorithm which simulates the process of natural selection of living organisms to find an approximate solution to a problem [31]. It is often used as an optimization method in solving the problems where not much is known about the objective function and the search space is large. GAs have been successfully applied to solve the different IR problems of automatic document indexing, query definition, document and term clustering, matching function learning, image retrieval, user profiles designing for web search, web page classification, design of agents for Internet searching [6].

This paper presents a state-of-the-art review of the studies carried out on the application of genetic algorithms in search query optimization. Section 2 briefs about the information retrieval system, Section 3 discusses about optimization of queries in information retrieval. Section 4 focuses on GA and query optimization in Information Retrieval. Section 5 presents a survey on the application of GAs in information retrieval for query optimization. Finally, the last section presents the concluding remarks.

II. INFORMATION RETRIEVAL

Information retrieval deals with the representation, storage, organization of, and access to structured and unstructured information items relevant to a user [3]. Information Retrieval System has three main components: Documentary database, Query Subsystem and Matching mechanism. Fig. 1 depicts a general Information Retrieval System (IRS) architecture [7].

A search system includes different phases [13]. Initially, the user conveys his information needs, which are abstract in nature, to the system. The needs are in the form of text from which keywords are extracted to formulate a query. The IR system transforms the query to enhance its usefulness. The documents in the database are then matched with the transformed query, by evaluating it against the internal document representation to decide about the relevancy of the document to the query. The retrieved documents are ranked or ordered based upon a ranking strategy. Few systems may try to improve the ranking depending on the feedback of the user about the relevancy of the retrieved documents.

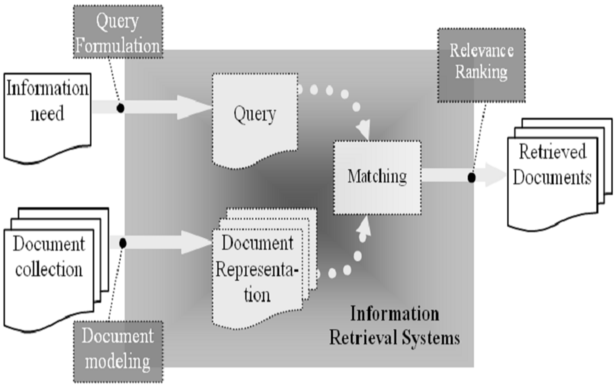

Fig. 1. A general IR system architecture

The structure of internal document representation, technique of document indexing, query language and document-query matching mechanism depends on theoretical Information Retrieval model underlying the system. Various IR models have been proposed over the years to predict about the relevancy. Some of the well known models are Boolean model, Vector space model (VSM) and Probabilistic model.Boolean model is the simplest, but finds exact matches onl. VSM and Probabilistic models yields better results b ranking the documents in order of their relevancy.

The quality of an IRS can be measured on the basis of the system efficiency, effectiveness, and numerous subjective aspects related to the user satisfaction. Two measures popularly used to evaluate the effectiveness of IR are: Precision and Recall [3]. Precision is defined as the ratio of number of relevant documents retrieved by a search to the total number of documents retrieved by that search. Recall is the rate between the number of relevant documents retrieved and the total number of relevant documents to the query existing in the database.

III. Query Optimization in Information Retrieval

An information need is the topic about which the user desires to know more, which is generally abstract in nature. A query is a formal statement of information needs which the user conveys to the computer. Few sample queries could be:

Q1: computer linguistic processing
Q2: metrics for software testing using genetic algorithm
Q3: upcoming apple products software

Information searching and browsing is iterative in nature, the users goes on finding information, learn something, and refining the query till relevant documents are located. The user query necessitates to be transformed into an effective form, using optimization techniques in order to improve the quality of recovered information, and to reduce the time taken for retrieval.

Query optimization is the process of selecting how to organize the work of answering a query so that the least total amount of work needs to be done by the system [15]. It involves reformulation of keywords by appropriate selection of a subset of search terms among a list of candidate terms. The subset selection will depend not only on the original terms but also on the overall query: an expansion that is correct in one query context may be incorrect in another [32]. The main goal in the query optimization is to maximize the subset of retrieved relevant documents and minimize the subset of retrieved non-relevant documents[27], thus improving the precision and recall values.

IV. Genetic Algorithms and Query Optimization in Information Retrieval

Genetic Algorithms (GAs) are adaptive and best fitted heuristic optimization techniques based on the evolutionary principles of natural and genetic selection, employing a population of individuals that undergo selection in the presence of variation-inducing operators, such as mutation and crossover. This process is iterative, and typically leads to better and fitter individuals. GAs offer powerful search mechanism and are suitable for information retrieval because of its robustness and quick search capabilities in large and complex search space. They manipulate a population of queries rather than a single query [25].

The main phases while designing a GA are:

- -Chromosome representation for the feasible solutions to the optimization problem.
- -Initial population of the feasible solutions.
- -A fitness function that evaluates each solution.
- -Genetic operators that generate a new population from the existing population.
- -Control parameters such as population size, probability of genetic operators, number of generations.

In Fig 2, a genetic model for Information Retrieval has been depicted. Query and documents both are encoded as chromosomes. An initial population of query is created. A fitness value attributed to each chromosome, is evaluated at each generation from the current population. The chromosomes are modified to form a new population, based on their fitness value. The query is sent to the information retrieval system and the query chromosome is matched against the document chromosomes for finding the relevant documents. If non relevant documents are found, then the query is reformulated. The query is reformulated until a relevant document is retrieved.

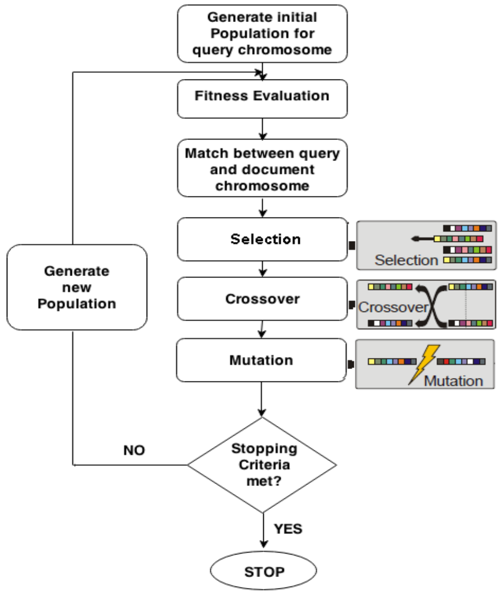

Fig. 2. Genetic Algorithm

V. LITERATURE REVIEW

Different researchers have combined GA with techniques of relevance feedback [1] [22] [28], reweighting document term indexing [30] , and fuzzy systems [4] [5] [26] to improve information retrieval process.

In [30], the authors proposed a GA to query optimization by reweighting the document terms without query expansion. They used a selection operator based on a stochastic sample, a blind crossover at two crossing points, and a classical mutation to renew the population of queries. This paper discusses an alternative approach for finding keywords of documents and then applying a genetic approach to adapt the weights of keywords. The experiments showed that the queries converge to their relevant documents after six generations.

In [28], the authors have considered optimization as a multimodal problem. They have used genetic technique of niching, knowledge-based operators and relevance feedback to improve upon the relevance of the documents. There is further scope of developing more effective merging formula and testing upon other niching techniques. Aly [1] and Radwan et. al. [22] discussed an adaptive method using GA to modify user query based on relevance judgements. The former uses cosine as fitness function, while latter developed a new fitness function which is the difference between terms weights of a given chromosome and the query vector. The data testbeds used were CISI, CACM and NLP. It calculated the similarity scores for any 100 queries of each collection. The algorithm show that application of GA result in better and effective queries with lesser number of terms. The weights of the query terms is adapted to give higher precision. The new fitness function by [22] proved to be quick and flexible than cosine similarity fitness function. They also discussed about order of complexity of the two fitness functions.

Many authors have used GA for optimizing Boolean queries. The authors have proposed Genetic Algorithm to optimize Boolean query consisting of not, and, or and xor [21]. The Boolean query is encoded in the form of tree prefix and the system is evaluated using precision and recall. They experimented using different collections of documents with variant number of words and documents, for upto 50 generations. The authors used GA to optimize Boolean query by focusing upon different types of mutation on terms over varied size of search space and compared two fitness measures viz precision and recall [12]. The results proved that recall is a better fitness function than precision to reach an optimized query in lesser number of generations.

In [26], the authors applied GA to improve upon vector & Boolean queries by modeling user needs through fuzzy models . They converted numerical weights into linguistic values to depict similarity & user feedback. They used F-score as a fitness measure and suggested scope for the task of adaptive implicit user modeling for vector or Boolean query improvement. Snásel et al. have proposed genetic and fuzzy oriented approach to IR optimization tasks, to help the user in determining useful search queries to fetch the most relevant information in her current context [27]. They proposed to simplify query optimization from multi-objective to a single-objective task, by using F-score fitness measure which combines precision and recall into one value. This study can be extended to calculate other fitness measures like fallout.

GA have also been successfully used in optimizing user query in Spanish [32], Chinese [11] and Arabic text collections [16] [18] [19]. Agüera and José used GA to reformulate user query to improve the search result by using a morphological thesaurus and applying stemming to obtain more relevant set of candidate terms for Spanish collection [32]. The population size used by the author was small, and function related to query performance prediction could also be considered. Horng & Yeh proposed an innovative approach to automatically retrieve keywords and then used genetic techniques to tune the keyword weights [11]. They demonstrated the effectiveness of their approach by comparing the results with a PAT-tree based approach. Nassar et al. proposed six different GA strategies combining different fitness functions and different mutation types to find the best strategy that can be applied upon the Arabic collection using the Boolean model[18]. The results of experiments showed that GA using mutation on the leaf node and the precision as a fitness function yielded the best performance. These strategies can be tested over other language text collections. The authors also performed query optimization over Arabic test collection based on vector space model [16] [19]. They created ten different GA strategies from combination of various crossover and mutation operators. They compared the results of these strategies using Dice and Inner Product similarity measures, and later using Cosine and Jaccard Coefficient. The strategy which employed one-point crossover operator, point mutation, and Inner Product similarity as a fitness function represent the best IR system in VSM for Arabic collection. Maitah et al. proposed an adaptive GA model for optimizing queries using VSM, Extended Boolean Model and Language Model for Arabic text collection [17]. They varied the values of crossover and mutation over the generations for faster and better results. In [29], the authors presented a comparison of various similarity coefficients to find best fitness value for web retrieved documents using Genetic Algorithm. They were restricted to result set of ten pages per query. There is a future scope of finding the optimal number of pages for achieving efficient crawling.

It was noted that there is a scope to apply GA to improvise information retrieval process to further develop a good user-centered system which boosts the user satisfaction and helps him in finding the system valuable. Table I shows a comparative analysis of diverse proposals that apply GA for query optimization in Information Retrieval.

TABLE I
DIVERSE PROPOSALS THAT APPLY GA FOR QUERY OPTIMIZATION IN INFORMATION RETRIEVAL

Different Prposals	**Retrieval Model**	**Fitness function**	**Operator used**	**Chromosomes**
Yang and Korfhage [30]	VSM	P=Relevant Docs Retreived -NonRelevantDocs Retreived -Relevant docs not retrieved	Selection two-point crossover random mutation	Each query vector is assigned random real weights corresponding to its terms.
Tamine and Boughanem [28]	Any	Guttman model: $Fitness(Q_u^{(s)}) = 1 + \frac{\sum_{dr \in Dr(s), dnr \in Dnr(s)} J(Q_u^{(s)}, dr) - J(Q_u^{(s)}, dnr)}{\sum_{dr \in Dr(s), dnr \in Dnr(s)} \left\| J(Q_u^{(s)}, dr) - J(Q_u^{(s)}, dnr) \right\|}$	Knowledge based operators Niching Virtual query One-point crossover	Each gene corresponds to an index term or concept and is assigned a real value.
Radwan et al.[22]	VSM	GA1= $\frac{\sum_{i=1}^{t} x_i . y_i}{\sqrt{\sum_{i=1}^{t} x_i^2 . \sum_{i=1}^{t} y_i^2}}$ GA2= $\sum_{i=1}^{t} x_i - y_i$	Selection Single-point crossover=0.8 Mutation=0.7 100 generations	The chromosomes is represented using binary values, then are converted to a real coding by using a random function.
A.A. Aly [1]	VSM	$\frac{\sum_{i=1}^{t} x_i . y_i}{\sqrt{\sum_{i=1}^{t} x_i^2 . \sum_{i=1}^{t} y_i^2}}$	Single-point crossover Mutation 100 generations	The chromosomes is represented using binary values.A random function is used to convert to a real encoding.
Owais et. al[20]	Boolean Model	Recall (E1) $E_1 = \frac{\sum_d [r_d \times f_d]}{\sum_d [r_d]}$ Precision (E2) $E_2 = \frac{\alpha \sum_d [r_d \times f_d]}{\sum_d [r_d]} + \frac{\beta \sum_d [r_d \times f_d]}{\sum_d [f_d]}$	Single-Point Crossover Mutation =0.2 50 generations	The Boolean query is represented in the form of tree, having all the terms at the leaf level and Boolean operators as internal nodes.
Husek et. Al [12]	Boolean Model	Recall (E1) $E_1 = \frac{\sum_d [r_d \times f_d]}{\sum_d [r_d]}$ Precision (E2) $E_2 = \frac{\alpha \sum_d [r_d \times f_d]}{\sum_d [r_d]} + \frac{\beta \sum_d [r_d \times f_d]}{\sum_d [f_d]}$	Point crossover =0.8 Different mutation rates =(0.1 - 0.5) 200 generations	The chromosome is a tree structure with indexing for all terms and all Boolean logical operators.
Snasel et. Al [27]	Boolean Model, Extended Boolean Model	Average of Precision & Recall, F-Score $P = \frac{\|REL \cap RET\|}{\|RET\|}$ $R = \frac{\|REL \cap RET\|}{\|REL\|}$ $F = \frac{2PR}{P+R}$	mutation = 0.2 crossover = 0.8 number of generations= 1000 population of 70 queries	Query represented as tree with weights assigned to each node.

Different Prposals	**Retrieval Model**	**Fitness function**	**Operator used**	**Chromosomes**
Snasel et. Al [26]	Fuzzy Extended Boolean Model, VSM	F-Score $F = \frac{2PR}{P+R}$	Linguistic inputs User modeling	Vector queries represented as pairs (t, w) where t is a keyword term and w. fitness adjustment based on linguistic inputs.
Agüera and José [32]	VSM	$(\cos\theta)^{1/2}$, $\cos\theta$, $\cos^2\theta$	Genetic stemming one-point crossover Mutation=0.25	Chromosomes are represented by fixed-length binary strings where each position corresponds to a query term.
Nassar et al. [18]	Boolean Model	$Recall = \frac{\sum_d [rd \times fd]}{\sum_d [rd]}$ $Precision = \frac{\alpha \sum_d [rd \times fd]}{\sum_d [rd]} + \frac{\beta \sum_d [rd \times fd]}{\sum_{id} [rd]}$	Single-point Crossover Mutation : • On term node • On operator • By insertion/ deletion of operator	Queries are represented as a tree with indexing for all terms and all Boolean logical operators.
Mashagba et al. [16]	VSM	Dice= $\frac{2\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sum_{i=1}^{t} w_{i,j}^2 + \sum_{i=1}^{t} w_{i,q}^2}$ Inner Product= $\sum_{k=1}^{t} (d_{ik} \bullet q_k)$	Combination of 1, 2 1)crossover: Five types • One-point • Restricted, • Uniform, • Fusion, • Dissociated 2)Mutation: Two types • Point • chromosomal	Random function is used to convert Binary representation to a real representation. Query and the feedback documents having terms with non-zero weights becomes the chromosome size.
Nassar et al. [20]	VSM	Jaccard Coefficient= $\frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sum_{i=1}^{t} w_{i,j}^2 + \sum_{i=1}^{t} w_{i,q}^2 - \sum_{i=1}^{t} w_{i,j} \times w_{i,q}}$ Cosine Similarity= $\frac{\sum_{i=1}^{t} x_i \cdot y_i}{\sqrt{\sum_{i=1}^{t} x_i^2 \cdot \sum_{i=1}^{t} y_i^2}}$	Same as Mashagba et al. (2011)	The chromosomes use a binary representation, and are converted to a real representation by using a random function.
Horng and Yeh[11]	VSM	Non-interpolated average precision $\mathrm{Avg}P = \frac{1}{\vert D \vert} \sum_{i=1}^{\vert D \vert} r(d_i) \sum_{j=1}^{\vert D \vert} \frac{1}{j}$	Weight-selection crossover, Natural crossover Inversion Mutation Local Search	Chromosome as a vector (w1, w2, w3, . . ., wn), where wi denotes the weight of the keyword ti between 0 and 1.
Maitah et al. [17]	VSM Extended Boolean Model Language Model	Cosine Similarity= $\frac{\sum_{i=1}^{t} x_i \cdot y_i}{\sqrt{\sum_{i=1}^{t} x_i^2 \cdot \sum_{i=1}^{t} y_i^2}}$ Horng & Yeh= $F = \frac{1}{\vert D \vert} \sum_{i=1}^{\vert D \vert} \left(r(di) \sum_{j=1}^{\vert D \vert} \frac{1}{j} \right)$	One-point crossover (0.5-1.0) Mutation (0.005-0.5) Adaptive GA	Chromosome is represented using binary coding. Number of genes is equal to terms with non-zero weights in query & feedback documents.

VI. CONCLUSION

This paper has dealt with the use of genetic algorithms in information retrieval for query optimization. To a great extent, the success of Information Retrieval System lies in user satisfaction which is highly dependent upon the quality of documents retrieved. GAs are characterized by higher probability of finding good solutions for large and complex problems like query optimization. The study exhibits that query optimization has been applied mainly on Boolean and Vector queries, the evaluation measure primarily being precision and recall. The genetic algorithm being used in most cases is generational Genetic Algorithm with Single-point Crossover and Point Mutation. In optimization of Boolean queries, the query mutation types include changing of node's weights, replacement, insertion and deletion of a compatible operator node. The genetic operators and other parameters can be varied to further improve the retrieval performance. Other types of crossover and mutation could be explored for further research in evolutionary improvement of search queries.

The table illustrates about fourteen diverese proposals on query optimization in Information Retrieval, out of which about 80% researchers employ single point crossover operator, with mutation probability ranging from 0.1 to 0.7 (Fig. 3).

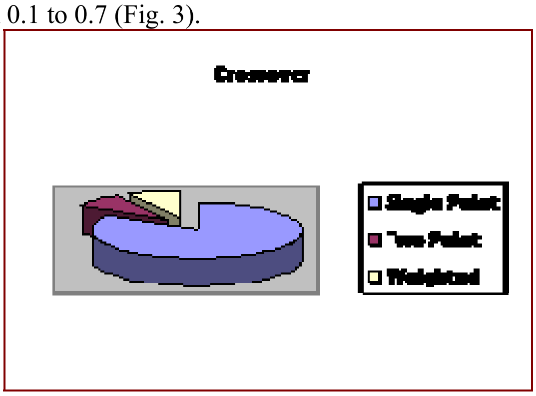

Fig. 3. Different Crossovers Used

Overall, the results of the studies categorically prove the applicability of genetic optimization algorithms in improving the Information Retrieval process. The paper presents diverse proposals on relevance of genetic algorithm in search query optimization which are promising and still developing areas of research. The results, thus far have, been very positive and encouraging and offer prospects for further research.

REFERENCES

[1] Aly, A. A. *Applying genetic algorithm in query improvement problem*. International Journal Information Technologies and Knowledge, 7(1): 309-316., 2007.

[2] Agbele, K., A. Adesina, D. Ekong and O.Ayangbekun. *State-of-the-art review on relevance of genetic algorithm to internet web search*. Applied Computational Intelligence and Soft Computing, 25, 2012.

[3] Baeza-Yates, R. and B. Ribeiro-Neto. *Modern Information Retrieval*, 463. ACM press, 1999.

[4] Chen, H.,1995. *Machine learning for Information Retrieval: Neural Networks, Symbolic Learning and Genetic Algorithms*, JASIS. 46(3) : 194-216.

[5] Cordón, O., F. Moya and C. Zarco, 2002. *A new evolutionary algorithm combining simulated annealing and genetic programming for relevance feedback in fuzzy information retrieval systems*. Soft Computing-A Fusion of Foundations, Methodologies and Applications. 6(5): 308-319, 2003.

[6] Cordón, O., E. Herrera-Viedma, C. López-Pujalte, M. Luque, M. and C. Zarco. *A review on the application of evolutionary computation to information retrieval*. International Journal of Approximate Reasoning. 34(2): 241-264, 2003.

[7] Crestani, F. and G. Pasi. *Soft Information Retrieval: Applications of fuzzy set theory and neural networks. Neuro-Fuzzy Techniques for Intelligent Information Systems*, Springer Verlag, Heidelberg, DE, 287-315, 1999.

[8] Drias, H. *Web information retrieval using Particle Swarm Optimization based Approaches*. In: Web Intelligence and Intelligent Agent Technology (WI-IAT) 2011, IEEE/WIC/ACM, IEEE, 1: pp. 36-39., 2011.

[9] Grossman D. A. and O. Frieder. *Information Retrieval: Algorithms and Heuristics*. Kluwer Academic Publishers, Norwell, MA, USA, 1998.

[10] Hansen, P., M.V. Poggi de Aragão and C.C. Ribeiro. *Hyperbolic 0–1 programming and query optimization in information retrieval*. Mathematical Programming, 52(1): 255-263, 1991.

[11] Horng, J. T., and C.C. Yeh. *Applying genetic algorithms to query optimization in document retrieval*. Information processing & management. 36(5), 737-759, 2000.

[12] Húsek, D., V. Snásel, S.S.J. Owais and P. Krömer. *Using Genetic Algorithms for Boolean Queries Optimization.* IMSA, 178-184, 2005.

[13] Jain, A. and S. V. Chande. *Heuristic Optimization of Queries in Information Retrieval.* Journal of IMS Group, 10(1): 96-102, 2013.

[14] Latha, K., S. Vigneshbabu, and R. Rajaram. *DETS: A Meta heuristic approach for document retrieval*. International Journal of Knowledge Engineering and Soft Data Paradigms 1(2): 139-150, 2009.

[15] Manning, C., P. Raghvan and H. Schutze. *An Introduction to Information Retrieval*. Cambridge University Press, 2008.

[16] Mashagba, E.A., F. A. Mashagba, and M. O. Nassar. *Query Optimization Using Genetic Algorithms in the Vector Space Model*. arXiv:1112.0052, 2011.

[17] Maitah, W., M. Al-Rababaa and G. Kannan. *Improving the Effectiveness of Information Retrieval System Using Adaptive Genetic Algorithm.* International Journal of Computer Science & Information Technology, 5(5):91-105, , 2013.

[18] Nassar, M.O., F. A. Mashagba and E. A. Mashagba. *Improving the User Query for the Boolean Model Using Genetic Algorithms*, IJCSI, 8(5): 66-70, 2011.

[19] Nassar, M. O., F. A. Mashagba and E. A. Mashagba. *Investigating Genetic algorithms to optimize the user query in the vector space model.* Australian Journal of Basic and Applied Sciences, 7(2): 47-53, 2013.

[20] Owais, S. S. J., P. Krömer, and V. Snášel. Evolutionary *Learning of Boolean Queries by Genetic Programming.* In: ADBIS Research Communications, 54-65, 2005.

[21] Owais, S.S.J., P. Krömer, V. Snásel. *Query optimization by Genetic Algorithms*. DATESO, 125-137, 2005.

[22] Radwan, A. A. A., B. A. A. Latef, A. M. A. Ali, and O.A. Sadek. *Using genetic algorithm to improve information retrieval systems*. World Academy of Science and Engineering Technology, 17(2), 6-13, 2006.

[23] Salton, G. and M.H. McGill. *Introduction to Modern Information Retrieval*, McGraw-Hill.

[24] Singhal, A. *Modern Information Retrieval: A Brief Overview,* Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, 24 (4) : 35–43, 2001, 1983.

[25] Simon, P. and S. S. Sathya. *Genetic algorithm for information retrieval.* In: Intelligent Agent & Multi-Agent Systems, IAMA 2009, IEEE, pp: 1-6, 2009.

[26] Snásel, V., P. Kromer, P. Musilek, H. O. Nyongesa, and D. Husek. *Fuzzy Modeling of User Needs for Improvement of*

Web Search Queries. In: Fuzzy Information Processing Society. NAFIPS. IEEE.446-451, 2007.

[27] Snásel, V., A. Abraham, S.S.J. Owais, J. Platos, and P. Krömer. *Optimizing Information Retrieval Using Evolutionary Algorithms and Fuzzy Inference System*, In: Proceedings of Foundations of Computational Intelligence (4):299-324, 2009.

[28] Tamine, L. and M. Boughanem. *Applying Heuristics to Improve A Genetic Query Optimisation Process in Information Retrieval,* ECIR, Darnstad, 2001.

[29] Thada, V. and V. Jaglan. *Comparison of Jaccard, Dice, Cosine Similarity Coefficient To Find Best Fitness Value for Web Retrieved Documents Using Genetic Algorithm*. IJIET, 2(4): 202–205, 2013

[30] Yang, J. J. and R. R. Korfhage. *Query Optimisation in Information Retrieval Using Genetic Algorithms*, ICGA, 1993.

[31] Goldberg, D.E.. *Genetic Algorithms in Search, Optimization and Machine Learning*. Pearson Education, New Delhi, ISBN-10: 0201157675, 2003.

[32] Pérez-Agüera, José R. *Using genetic algorithms for query reformulation*. BCS IRSG Symposium: Future Directions in Information Acces, 2007.