International Journal of Soft Computing 5 (3): 155-163, 2010
ISSN: 1816-9503
© Medwell Journals, 2010

A Novel Approach Integrating Ranking Functions Discovery, Optimization and Inference to Improve Retrieval Performance

Kehinde K. Agbele, Ademola O. Adesina, Henry O. Nyongesa and Ronald Febba
Soft Computing and Intelligent Systems Research Group,
Department of Computer Science, University of the Western Cape,
Private Bag, X17, Bellville 7535, South Africa

Abstract: The significant roles play by ranking function in the performance and success of Information Retrieval (IR) systems and search engines cannot be underestimated. Diverse ranking functions are available in IR literature. However, empirical studies show that ranking functions do not perform constantly well across different contexts (queries, collections, users). In this study, a novel three-stage integrated ranking framework is proposed for implementing discovering, optimizing and inference rankings used in IR systems. The first phase, discovery process is based on Genetic Programming (GP) approach which smartly combines structural and contents features in the documents while the second phase, optimization process is based on Genetic Algorithm (GA) which combines document retrieval scores of various well-known ranking functions. In the 3rd phase, Fuzzy inference proves as soft search constraints to be applied on documents. We demonstrate how these two features are combined to bring new tasks and processes within the three concept stages of integrated framework for effective IR.

Key words: Ranking function, information retrieval, evolutionary techniques, fuzzy inference system, data fusion method

INTRODUCTION

The goal of a ranked retrieval system is to manage a large collection of text documents and to order documents for a user based on the estimated relevance of the documents to the user's information need. Information Retrieval (IR) field is undergoing amazing development and change due to advances in Information and Communication Technology (ICT) and computational techniques. Information stored by one person in a data repository is preferred to be retrieved by another. Emphasizing the WWW as the data repositories are used for continuing information both in space and time.

At the moment a search engine has become useful by many people to discover information online that will assist them formulate better knowledgeable decisions. According to searchenginewatch.com, major search engines such as Google and Yahoo take delivery of millions of search request per day. This fact obviously exemplifies the significance of search engines in the daily life. However, the understandings with search engines demonstrate that their potentials of getting back useful and relevant results are not always pleasing. Hence, there is need to refine the search query several times and search from beginning to end a long list of document collections only to discover a few of them relevant. To address the issue of improving retrieval performance, diverse techniques have been used by information retrieval researchers (Gao *et al*., 2002; Salton, 1989).

There are four basically subsystem components of information retrieval system: Documents, Queries, Matching functions and Ranking functions. A document collection consists of documents about many different topics. Documents are represented in a form that basically uses vector space model (Salton, 1989) that can easily be employed for the ranking functions. There is need to translate user's information needs into queries for the system to process. Query formatting depends on basic model of retrieval used viz: Vector Space Model (Salton, 1989), Fuzzy retrieval models (Bordogna and Pasi, 1993), Probabilistic models (Robertson and Jones, 1976), Boolean models (Bookstein, 1985) and models based on artificial intelligence techniques (Chen *et al*., 1998). So far, previous ranking function discovery efforts have centered on the Vector Space Model (VSM) in which all

Corresponding Author: Kehinde K. Agbele, Soft Computing and Intelligent Systems Research Group,
Department of Computer Science, University of the Western Cape, Private Bag, X17, Bellville 7535,
South Africa

documents and queries are represented as vectors and the same term weighting strategy used in a ranking function is applied to all terms in a document. Diverse IR experiment evaluations and TREC competitions show that ranking functions based on VSM have performed very well (Harman, 1993; Salton, 1989).

The documents are presented to users to rate as either relevant or non-relevant to his/her information needs. In IR system performance criteria like precision and recall are the two most widely used measures of retrieval performance in meeting users information needs. Recall is the ratio of the number of the relevant documents available in the document collection. Precision is defined as the ratio of the number of relevant retrieved documents to the total number of retrieved documents. A common method for query formulation is called the relevance feedback (Salton, 1989) and allows a user to interactively express information requirements by modifying successive query inputs.

The performance of a search engine can be affected by factors like indexing and query representation etc. (Lancaster and Warner, 1993). But ultimately, it is affected by the ranking function which is used to rank documents according to its match with a user's query. In manipulating the documents and queries to improve retrieval performance, researchers have broadly looked at how to achieve this task (Gao *et al*., 2002; Kraft *et al*., 1997; Horng and Yeh, 2000).

In this study we focus the attention on discovering, optimizing and drawing inference for the ranking functions. Basically in the web scenario, ranking functions exploit three characteristics of the documents: the contents of the documents, the links to the documents and the structure of the document. The content based ranking functions (Robertson *et al*., 1996) make wide usage of diverse lexical/syntactical statistics of words in a document collection: tf, df, dl, etc. for ranking purposes. Link based ranking functions utilize web interconnection to assist boost the ranking performance by identifying those authoritative pages which are highly certified by others on popular topics (Kleinberg, 1999).

Structure based ranking functions exploit the structural properties in documents by assigning weights to words appearing in different structural positions such as Title, Header, Anchor and use those weighting heuristics to improve ranking performance. Various ranking functions seek to combine the evidence at the content, link and structure levels as evidenced in the second TREC web track competition (Hawking and Craswell, 2002). In the TREC competition it was obvious that using link information unaided does not provide much help in performance improvement as compared to using content information alone. Also, the ranking functions based on content alone are still very successful. For example, Okapi (Singhal *et al*., 1996), a ranking function based on content unaided was found very thriving.

There is some prior research in using GP for ranking function discovery (Fan *et al*., 2004a, b) and using GA for ranking fusion (Billhardt *et al*., 2003). There is fuzziness and instinct in human mind. It involves the means of communication. Estimation and instinct are present. These facts influence both-information content of the documents and search request formulations. Moreover, the document content is described only in a rough, imperfect way (Bordogna and Pasi, 2001). To the understanding there is no research combining these three stages into a logical integrated framework.

Novel nonlinear optimization is known to be associated with GP. However, it remains to be searched whether the novel ranking functions discovered by GP can be fused later with other well-known ranking functions by GA to further improve ranking function performance. IR is seen as fuzzy multi-criteria decision making in the presence of vagueness within the fuzzy set framework.

We think that these three flows of ranking function improvement research can be integrated yielding improved retrieval performance. In this study, an integrated three-stage framework for improving retrieval performance is proposed. In the 1st phase called searching phase that make use of the structural information in documents along with the content information in them to discover new ranking functions. GP is use for such a discovery.

The 2nd phase called optimization phase combines the information provided by well-known ranking functions including the ones discovered by GP using an optimization technique like GA to further improve retrieval performance. In the 3rd phase called the deduction phase which deploys rule based on fuzzy ranking of the documents collection according to the level of their conformity to the soft search criteria specified via user queries.

THEORETICAL FOUNDATION

Purposely, first review the Vector Space Model (VSM) which is the theoretical model upon which the integrated framework is based. Then we will review related research in data fusion technique as applied to IR and IR that uses GP, GA and Fuzzy principles.

Vector space model: The VSM is chosen to be the theoretical foundation for these reasons: The VSM is a theoretically well-grounded model due to ease of interpretation and can be easily interpreted from a geometric perspective (Jones and Furnas, 1987). For example each document and query is placed in an n dimensional space where its properties can be studied using geometrical similarity. As a result of great success in performance evaluations, the VSM has been one of the most successful models in various performance evaluation studies (Harman, 1993; Salton and Buckley, 1988) and most existing search engines and information retrieval systems are designed based on it.

More purposely, both documents and user queries are represented as vectors in the VSM. Suppose there are total t index terms in an entire collection, a given document D and query Q can be represented as follows:

$$D=(w_{d1},w_{d2},w_{d3}\dots w_{dt})$$
$$Q=(w_{q1},w_{q2},w_{q3}\dots w_{qt})$$

where, wd_i, wq_i (for i = 1-t) are term weights assigned to different terms for the document D and query Q, respectively. The similarity between a query and a document can be calculated by the widely used Cosine measure (Salton and Buckley, 1988):

$$\text{Similarity}(Q,D)=\frac{\sum_{i=1}^{t} wq_i * wd_i}{\sqrt{\sum_{i=1}^{t}(wq_i)2 * \sum_{i=1}^{t}(wd_i)2}} \qquad (1)$$

Documents are then ordered by the decreasing values of this measure called Retrieval Status Value (RSV), is calculated for each document in the collection and the documents are ordered and presented to the user in the decreasing order of RSV for final ranking. There are various features available in the VSM to compute the term weights: wd_i, wq_i (for i = 1-t). One of the most widely used features for term weighting is term frequency (tf) which measures the number of times a term appears in a document or query. Another commonly used feature is the inverse document frequency (idf) which can be calculated by log (N/df) where N is the total number of documents in a text collection and df is another feature that measures the number of documents in which a term has appeared in an entire document collection. More features used in term weighting can be found by Salton (1989) and Salton and Buckley (1988). These features can also be combined to generate a wide range of new composite weighting features, e.g., tf* idf, etc.

Equation 1 suggests that to discover a good ranking function, we need to discover the optimal way of assigning weights to document and query keywords. Traditional VSM in the functional space combines a set of these weighting features such as tf, df, idf etc. It does not typically take into account the structural information within documents. If consider these weighting features to include the structural/position information such as Anchor, Title, Abstract and Body. Expanded set of features including tf_{anchor}, tf_{title}, $tf_{abstract}$, tf_{body} can get. The theoretical foundation serving Eq. 1 can still be applied to the structural context.

Equation 1 as the theoretical foundation for this study. In the first phase of the framework, we seek to discover new ways of leverage structural information in assigning weights to document and query terms to improve the overall ranking performance.

Related work on combining ranking functions for optimization: Data fusion technique has been basically applied in IR in the context of combining similarities obtained from different query representations and also on combining query representations themselves (Belkin *et al*., 1995). Hence, this involves the 2nd phase of the framework. Successful combination of diverse Boolean query formulations bring about improved retrieval performance. Various attempts have been made on ranking function optimization in IR literature (Fox and Shaw, 1994) in his finding used sum of individual similarities to combine retrieval results from diverse specialists. Bartell *et al*. (1994) concluded that combinations of three different experts on two test collections and established that an optimized combination performed better than any individual systems. Lee (1997) used ranks instead of similarity to extend the research of Fox (Bartell *et al*., 1998) used numerical methods to optimize only the parameters involved in a standard inner product measure. In addition, Savoy *et al*. (1996) combined okapi probabilistic model with diverse vector space schemes and used a heuristic to determine the best retrieval expert for a given query. Vogt and Cottrell (1999) used linear combinations of three experts (a binary scheme, a tf-idf weighted scheme and latent semantic indexing) to determine a set of parameters. Their method worked well on training set of documents but did not generalize well to hidden text documents.

Genetic and Fuzzy logic-based approaches in IR: GA (Holland, 1992) and GP (Koza, 1992) are artificial intelligence search algorithms based on evolutionary theory. They represent the solution to a problem as a chromosome (or an individual) in a population pool. They evolve the population of chromosomes in successive generations by following the genetic modification operations such as reproduction, crossover and mutation to discover chromosomes with better fitness values. The goal of a GA is the optimization of a fitness function which expresses the performance of a specific solution. As a result of powerful global searching capability in a high-dimensional space, both GA and GP have been used to solve a wide range of hard optimization problems. GA's are basically used to solve difficult parameterized nonlinear optimization problems while GP is basically used to approximate or discover complex, nonlinear functional relationships (Hawking and Craswell, 2002). Fuzzy logic, as a framework describing formally the concepts of vagueness provides interesting extensions to the area of IR. User friendly and flexible advanced information retrieval system should be able to offer user interface for non experienced users allowing natural deployment of fuzzy logic in user system interaction for more effective IR (Kraft *et al*., 1997). Diverse evolutionary algorithms were proposed at multiple stages of the IR process. Fan *et al*. (2004a, b) introduced genetic ranking function discovery framework. Nyongesa and Maleki-Dizaji (2006) used evolutionary interactive learning for user modeling. We now advance to present the framework for ranking function discovery, optimization and inference.

PROPOSED THREE-STAGE INTEGRATED FRAMEWORK FOR RANKING

A three-stage integrated framework to study the problem of ranking function discovery, optimization and inference in a web search context. This approach will help us in ranking function design by capably leveraging both the content and structural information entrenched in the documents. Efficiency is a central concern for any method in IR because of the number of documents involved in the task and as well as the number of features of each document.

Evaluation studies (Singhal *et al*., 1996) have shown that no single ranking function performs best for all contexts of document collections and queries on the use of ranking functions. The 1st phase of the framework searches a variety of hints available in content and structural information about the documents and the queries to discover new ranking functions, the 2nd phase smartly combines the facts obtained from these newly discovered ranking functions as well as from well-known open ranking functions to yield better retrieval performance while the 3rd phase shows as a form of suitable modeling for handling imprecision to governs system behavior. We apply GP for the first discovery phase, applying GA for the second optimization phase while applying fuzzy logic for the third inference phase. We have used GP in the first phase instead of GA while discovering new functions following (Koza, 1992) argument. In the 2nd phase we have used data-fusion techniques to combine the evidence obtained from various well- known ranking functions including the ones discovered in the 1st phase. This combination is done by weighing the score attained by each ranking function. GA's have shown to be very useful for such fusion (Zobel and Moffat, 1998; Fan *et al*., 2006a, b). Thus, we use GA in the second phase. Fuzzy set in the 3rd phase is used to govern the system behavior. The framework is shown in Fig. 1 and each of the phases is described in the following:

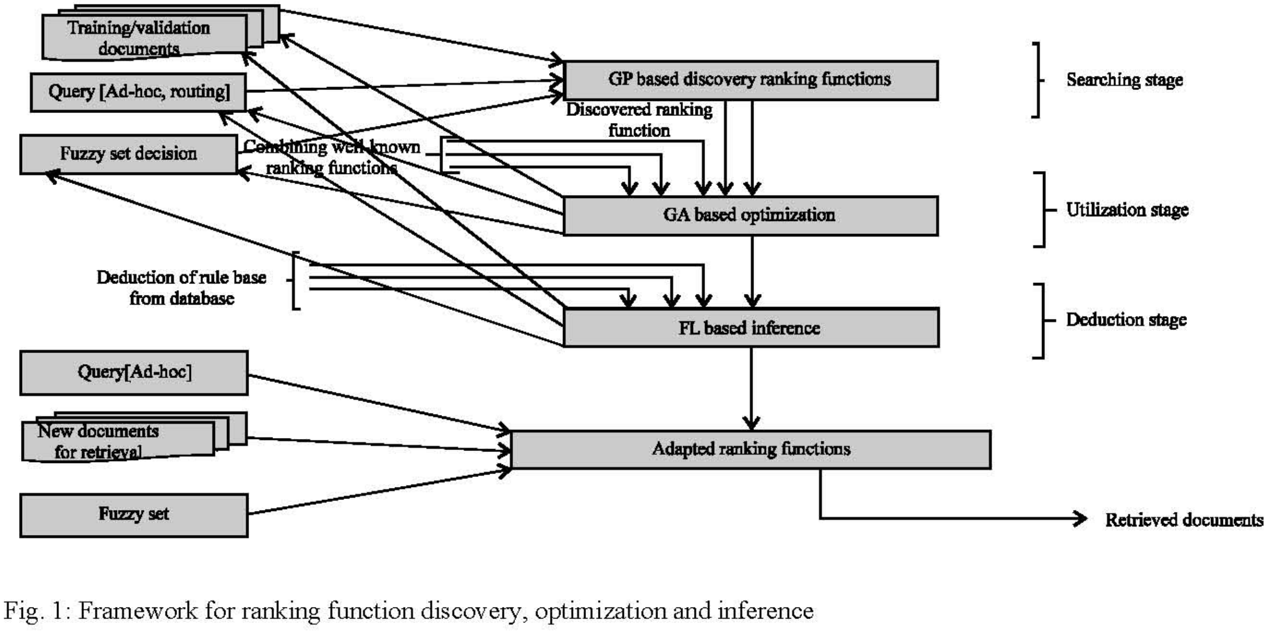
Training/validation documents
Query [Ad-hoc, routing]
Fuzzy set decision
Combining well known ranking functions
GP based discovery ranking functions
Discovered ranking function
Searching stage
GA based optimization
Utilization stage
Deduction of rule base from database
FL based inference
Deduction stage
Query[Ad-hoc]
New documents for retrieval
Fuzzy set
Adapted ranking functions
Retrieved documents

Fig. 1: Framework for ranking function discovery, optimization and inference

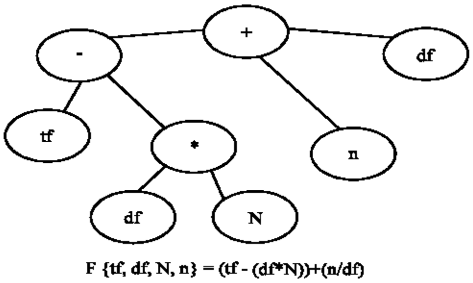

Fig. 2: Sample tree representation for a ranking function

Ranking function discovery based on Genetic Programming (GP): This approach was developed recently by (Koza, 1992). Koza suggests that the desired program should evolve itself during the evolution process. In other words, we should rather search the space of possible computer programs for the most fit. Genetic Programming (GP) is an inductive learning technique designed following the principles of biological inheritance and evolution (Koza, 1992) which provides a way to run a search.

In GP, each potential solution is called an individual in a population. An individual in GP systems is basically represented using a tree data structure as shown in Fig. 2 which combine these features. GP works by iteratively applying genetic operators, such as reproduction, crossover and mutation to a population of individuals to create more diverse and better performing individuals in subsequent generations.

GP based discovery of the ranking function is the 1st phase in the framework. Training and validation set of documents; Ad-hoc or routing queries as well as fuzzy decision serve as input to this discovery process.

Ad-hoc queries are used when we want to discover ranking function applicable to any queries issued to the system while routing queries are used when we are interested in discovering a query specific ranking function for each individual query. According to (Fan *et al.*, 2006a, b), the foundation of this discovery framework arose from his previous research done.

In that research only the content information from the documents and queries is exploited. We enhance the research by including the structure based information in the documents and queries.

Adding structure information to the retrieval process with Okapi ranking function, we should expect retrieval performance to be enhanced considerably (Fan *et al.*, 2006a, b). Moreover, Fan *et al.*, 2004a, b framework is enhanced by discovering ranking functions for ad-hoc queries as well. Table 1 shows some vital components of the model as well as content and structure based information along with their descriptions. The search space is a hyperspace of valid programs which can be viewed as a space of rooted trees which combines these features in such a way as to improve the retrieval performance.

Table 1: Terminals used in the GP X is used to stand for different parts of a document

Terminals used	Statistical meaning
tf_X	Number of times the term appears in part X of the document
tf_avg_X	Average tf in the part X of the document
tf_max_X_Col	Maximum tf_X in the entire document
df_X	Number of documents in the collection the term appeared in the part X
df_max_X	Maximum df_X for a given query
N	Number of documents in the entire text collection
Length_X	Length of a document part X
Length_avg_X_Col	Average length of part X in the entire collection
n	Number of unique terms in a document

Table 2: Essential components of the GP system

GP parameters	Meaning
Terminals	Leaf nodes in the tree data structure
Functions	Non-leaf nodes used to combine the leaf nodes
Fitness function	The objective function that need to be optimized
Reproduction	Genetic operators used to copy fit solutions from one generation to another
Crossover	Genetic operators used to introduce diversity in the population

Besides, Fig. 2 shows these features. The discovery of an optimal tree representing the ranking function is essentially the searching, utilization and deduction phase of the model.

GP is used for discovering such a tree because of these reasons. First, GP can be used to optimize any type of fitness function. GP is suited for this task as it does not require the fitness function to be continuous or differentiable.

Finally it has been shown empirically that solutions discovered by GP are basically better than those discovered by other heuristic algorithms which could automatically learn the optimal ranking function for the given context and very useful for nonlinear function discovery (Koza, 1992).

Table 2 shows some components of the GP system. An individual in the population is expressed in terms of a tree which represents one possible ranking function. A population in a generation consists of P such tree. Each tree is composed of functions and terminals appropriate to the particular domain.

We use features shown in Table 1 and real-valued number as terminals. The following are the five major steps to follow in using GP for a particular problem:

- Selection of terminals
- Selection of a function
- Identification of the objective function
- Selection of parameters of the system
- Selection of the termination condition

Hence these functions were used in the studies: +,-, * and /. P-Avg is used as the fitness function also called evaluation function which is defined in Eq. 2:

$$P-\text{Avg}=\sum_{i=1}^{|D|}\left(r(d_i)*\left(\sum_{j=1}^{i}r(d_j)/i\right)\right)\Big/\text{TRel} \qquad (2)$$

Where r (d_i), {x, y} is the relevance score assigned to a document, it is assigned y if the document is relevant for y≤ i≤|D| and x otherwise for i>|D|. |D| is the total number of retrieved document. TRel is the total number of relevant documents for the query. P-Avg is the standard performance measure used in retrieval studies because it takes into account not just how many relevant documents are retrieved but also the positions at which they are retrieved (the more relevant documents at the top the better the P-Avg score). Hence, it combines both precision and recall in one single measure. Reproduction is a genetic operators used to copy fit solutions from one generation to another. The process involves a situation whereby top two parent's trees (in terms of fitness) are selected for crossover and they exchange sub-trees to form trees for the next generation. This is randomly selected.

Ranking function optimization based on Genetic Algorithm (GA): GA is proposed to study the newly discovered ranking functions by the GP process and combine with the evidence from the existing ranking functions to further improve the retrieval performance. It is evident that diverse ranking functions give varying significance to diverse features in the documents and queries and thus yield better results. The 2nd phase of the framework tries to provide solution to the issue raised above. Due to the inadequate scope, the optimization stage in the frame work is based on the research done by Fan where only the content information is exploited. However it did not use any structural information in documents. The optimization of ranking functions was done at the individual query level for the routing task. In this research we improve upon their research by integrating structural information in documents. We also had shown how the framework has been adapted using feedback information.

From Fig. 1, some of the newly discovered ranking functions from the GP discovery phase as well as other existing well-known ranking functions and fuzzy set will be used as input ranking functions for the optimization process. The optimization problem for the GA is shown in Fig. 3 where there are n different ranking functions (including the ones discovered by the first phase GP). For a given query, each ranking function assigns a retrieval score to each document in the document collection. These retrieval scores are weighed with a weight w (from w_1 to w_n, respectively) and linearly combine these weighted scores. The documents in the collection are ordered in the decreasing order of this weighted score and at the top are the number of documents a user is willing to see (retrieved documents for the user). The user judges these documents as either relevant or non-relevant for his/her information needs. Based on these judgments the retrieval performance of the system in is calculated terms of P-Avg (Eq. 2).

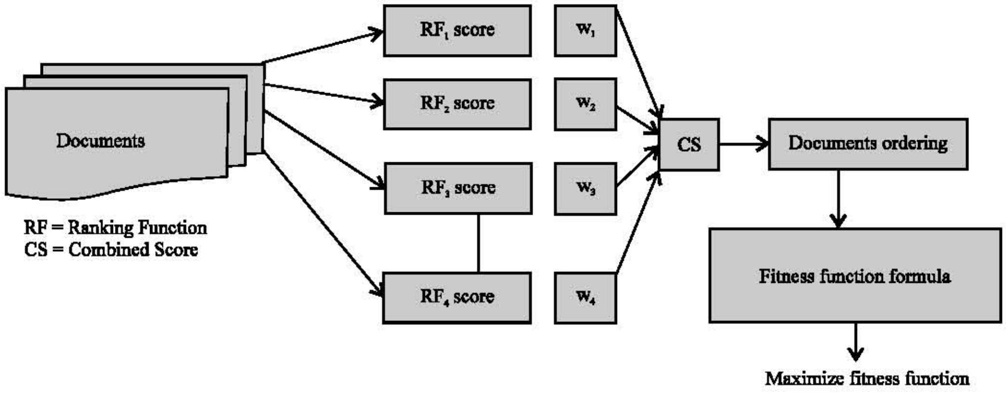

Fig. 3: Ranking function genetic algorithm optimization problem

The optimization problem is to maximize the P-Avg performance measure subject to proper assignment of the weights w_1-w_n. GA to is utilized do the assignment of these weights. GA performs a multi-directional search by maintaining a population of potential solutions and encourages information formation and exchange between these directions. The search space is typically infinite and the objective function of Eq. 2 is discrete in nature. Genetic Algorithm's provide a technique useful for fining approximate solutions to optimization and search problems.

A set of ranking functions including the ones discovered by prior GP discovery phase and associated weights are expected at the end of the optimization phase.

Fuzzy inference values in information retrieval: We propose a Fuzzy Inference System (FIS) which focus on the ability of fuzzy logic suitable for modeling natural language (Bordogna and Pasi, 2001) and to govern system behavior. When modeling information and requests containing vagueness or imprecision this introduce significant improvements to the search results. Information retrieval optimization based on knowledge of previous user search activities and fuzzy softening of both search criteria and information models. We introduce fuzzy oriented approach to these tasks with the goal to determine useful search queries describing documents relevant to user's area of interest as deducted from previous searches as a tool helping user to fetch the most relevant information in his or her current context.

Fuzzy concepts affect most phases of IR process. They are deployed during document indexing, query formulation and search request evaluation. Information retrieval is seen as fuzzy multi-criteria decision making in the presence of vagueness. In general, document is interpreted as a fuzzy set of document descriptors and queries as a composite of soft search constraints to be applied on documents. Document-query evaluation process is based on fuzzy ranking of the documents in documentary collection according to the level of their conformity to the soft search criteria specified via user queries. The document-query matching has to deal with the uncertainty arising from the nature of the fuzzy decision making and from the fact that user information needs can be recognized in terpreted and understood only partially. Fuzzy techniques support different grades of document-query relevance, cut inaccuracies and oversimplifications happening during document indexing and introduce the concepts of vagueness in query language (Kraft *et al*., 1997). In the fuzzy enabled IR framework, Linguistic variables such as probably or it is possible that can be used to declare the partial preference about the truth of the stated information. The interpretation of linguistic variables is then among the key phases of query evaluation process. The decision process performed by the query evaluation mechanism computes the degree of satisfaction of the query by representation of each document. This degree called Retrieval Status Value (RSV), is considered as an estimate of the relevance of the document with respect to the query. RSV = 1 corresponds to maximum relevance and RSV = 0 denotes no relevance (Bordogna and Pasi, 2001; Kraft *et al*., 1997). Automated text indexing deals with imprecision since the terms are not all fully significant to characterize the document content and their statistical distribution does not reflect their relevance to the information included in the document. Their significance depends also on the context in which they appear and on the unique personality of the inquirer. A flexible IR system should be designed to afford detailed and rich representation of documents, sensibly interpret and evaluate soft queries and hence offer efficient information retrieval service in the condition of imprecision (Kraft *et al*., 1997).

CONCLUSION

Information retrieval systems have gone over an intensive evolution process to satisfy the increasing needs of growing data bases. In their mature form they are still present in the heart of internet search engines as one of the key communication focal points of the society. Information search is one of the most vital e-activities. Despite their superior feature, the IR system needs modification and advancement in order to achieve better performance and provide users with relevant information needs. The achievement of better retrieval performance, evolutionary search algorithms and fuzzy set techniques are often a challenge. In this study, a novel ranking integrated framework is offered for using genetic programming, genetic algorithms and Fuzzy logic in the field of information retrieval to discover new ranking function; optimize the well-known existing ones and as well to model the system behavior to handle imprecision.

The first phase of the proposed framework uses GP to discover novel ranking functions. Both the content as well as structural information in the documents are used to discover such functions. In order to improve the retrieval performance further we integrate the second phase of optimization in the framework. It uses the scores assigned by individual ranking functions to the documents and assign weights to these scores. Hence, these set of weights are optimized using GA's. To govern the system behavior, the third phase of inference in the framework is integrated. Fuzzy set framework has been proved as suitable formalism for modeling and handling vagueness. The deployment of fuzzy techniques in IR has brought improvement of IR effectiveness and therefore, increases user information satisfactions. User feedback about the relevance of documents retrieved can be used to fine-tune the starting ranking function, starting weights and starting fuzzy sets.

The deployment of genetic programming and genetic algorithms for query optimization and fuzzy set techniques for better document modeling brings a significant contribution to the ultimate goal of web search. Thus, improves retrieval performance efficiency.

ACKNOWLEDGEMENT

The researchers wish to thank and appreciate constructive remarks from colleagues in the research group (Soft Computing and Intelligence Systems) during the earlier presentation of this research.

REFERENCES

Bartell, B., G. Cottrell and R.K. Belew, 1994. Automatic combination of multiple ranked retrieval systems. Proceedings of the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, Dublin, Ireland, July 3-6, Springer-Verlag, New York, USA., pp: 173-181.

Bartell, B.T., G. Cottrell and R.K. Belew, 1998. Optimizing similarity using multi-query relevance feedback. J. Am. Soc. Inform. Sci., 49: 742-761.

Belkin, N.J., P. Kantor, E.A. Fox and J.A. Shaw, 1995. Combining the evidence of multiple query representations for information retrieval. Inform. Process. Manage. Int. J., 31: 431-448.

Billhardt, H., D. Borrajo and V. Maojo, 2003. Learning retrieval expert combinations with genetic algorithms. Int. J. Uncertainty, Fuzziness Knowledge-Based Syst., 11: 87-113.

Bookstein, A., 1985. Probability and fuzzy set applications to information retrieval. Ann. Rev. Inform. Sci. Technol., 20: 117-151.

Bordogna, G. and G. Pasi, 1993. A fuzzy linguistic approach generalizing Boolean information retrieval: A model and its evaluation. J. Am. Soc. Inform. Sci., 44: 70-82.

Bordogna, G. and G. Pasi, 2001. Modeling Vagueness in Information Retrieval. Springer-Verlag, New York, pp: 207-241.

Chen, H., Y. Chung, M. Ramsey and C. Yang, 1998. A smart itsy bitsy spider for the web. J. Am. Soc. Inform. Sci., 49: 604-618.

Fan, W., D. Gordon and P. Pathak, 2004a. A generic ranking function discovery framework by genetic programming for information retrieval. Inform. Process. Manage., 40: 587-602.

Fan, W., M. Gordon and P. Pathak, 2004b. Discovery of context-specific ranking functions for effective information retrieval using genetic programming. IEEE Trans. Knowledge Data Eng., 16: 523-527.

Fan, W., M. Gordon and P. Pathak, 2006a. An integrated two-stage model for intelligent information routing. Decision Support Syst., 42: 362-374.

Fan, W., M. Gordon and P. Pathak, 2006b. On linear mixture of experts approaches to information retrieval. Decision Support Syst., 42: 975-987.

Fox, E.A. and J.A. Shaw, 1994. Combination of multiple searches. Proceedings of the 2nd Text Retrieval Conference (TREC-2), (TRC'94), NIST., pp: 243-252.

Gao, J., G. Cao, H. He, M. Zhang, J. Nie, S. Walker and S.E. Robertson, 2002. TREC-10 web track experiments at MSRA. Proceedings of the 10th Text Retrieval Conference, (TRC'02), NIST Special Publication, pp: 384-392.

Harman, D.K., 1993. Overview of the first text retrieval conference (TREC-1). Proceedings of the 1st Text Retrieval Conference, (TRC'93), NIST Special Publication, pp: 1-20.

Hawking, D. and N. Craswell, 2002. Overview of the TREC-2001 web track. Proceedings of the 10th Text Retrieval Conference, (TRC'01), NIST., pp: 61-67.

Holland, J.H., 1992. Adaptation in Natural and Artificial Systems. 2nd Edn., MIT Press, Cambridge, MA, pp: 211.

Horng, J. and C. Yeh, 2000. Applying genetic algorithms to query optimization in document retrieval. Inform. Process. Manage., 36: 737-759.

Jones, W.P. and G.W. Furnas, 1987. Pictures of relevance: A geometric analysis of similarity measures. J. Am. Soc. Inform. Sci., 38: 420-442.

Kleinberg, J.M., 1999. Authoritative sources in a hyperlinked environment. J. Assoc. Comput. Machinery, 46: 604-632.

Koza, J.R., 1992. Genetic Programming: On the Programming of Computers by Means of Nature Selection. MIT Press, Cambridge, MA., ISBN: 0-262-11170-5.

Kraft, D., F. Petry, B. Buckles and T. Sadasivan, 1997. Genetic Algorithms for Query Optimization in Information Retrieval: Relevance Feedback. In: Genetic Algorithms and Fuzzy Logic Systems: Soft Computing Perspectives, Sanchez, E., T. Shibata and L.A. Zadeh (Eds.). World Scientific Publishing Co. Pte. Ltd., London, pp: 155-173.

Lancaster, F.W. and A.J. Warner, 1993. Information Retrieval Today. Information Resources Press, USA., ISBN-13: 978-0878150649, pp: 341.

Lee, J., 1997. Analysis of multiple evidence combination. Proceedings of 12th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Philadelphia, Pennsylvania, United States, July 27-31, ACM, New York, pp: 267-276.

Nyongesa, H.O. and S. Maleki-Dizaji, 2006. User modeling using evolutionary interactive reinforcement learning. Inform. Retrieval, 9: 343-355.

Robertson, S.E. and K.S. Jones, 1976. Relevance weighting of search terms. J. Am. Soc. Inform. Sci., 27: 129-146.

Robertson, S.E., S. Walker, K.S. Jones, M.M. Hancock-Beaulieu and M. Gatford, 1996. Okapi at TREC-4. Proceedings of the 4th Text Retrieval Conference, (TRC'96), NIST Special Publication, pp: 73-96.

Salton, G. and C. Buckley, 1988. Term-weighting approaches in automatic text retrieval. J. Inform. Proc. Manage., 24: 513-523.

Salton, G., 1989. Automatic Text Processing: The Transformation, Analysis and Retrieval of Information by Computer. Addison-Wesley Publishing Co., MA.

Savoy, J., M. Ndarugendamwo and D. Vrajitoru, 1996. Report on the TREC-4 experiment: Combining probabilistic and vector space schemes. Proceedings of the 4th Text REtrieval Conference (TREC-4), Oct. 1, National Institute of Standards and Technology, Gaithersburg, MD, pp: 537-548.

Singhal, A., G. Salton, M. Mitral and C. Buckley, 1996. Document length normalization. Inform. Process. Manage., 32: 619-633.

Vogt, C. and G. Cottrell, 1999. Fusion via a linear combination of scores. Inform. Retrieval, 1: 151-173.

Zobel, J. and A. Moffat, 1998. Exploring the similarity space. ACM. SIGIR. Forum, 32: 18-34.