Improving pseudo relevance feedback based query expansion using genetic fuzzy approach and semantic similarity notion

Pragati Bhatnagar and Narendra Pareek

Journal of Information Science published online 19 May 2014
DOI: 10.1177/0165551514533771

Article

JIS

Journal of Information Science
1–15

DOI: 10.1177/0165551514533771
jis.sagepub.com
SAGE

Improving pseudo relevance feedback based query expansion using genetic fuzzy approach and semantic similarity notion

Pragati Bhatnagar
Department of Computer Science, M.L. Sukhadia University, India

Narendra Pareek
Department of Computer Science, M.L. Sukhadia University, India

Abstract
Pseudo relevance feedback-based query expansion is a popular automatic query expansion technique. However, a survey of work done in the area shows that it has a mixed chance of success. This paper captures the limitations of pseudo relevance feedback (PRF)-based query expansion and proposes a method of enhancing its performance by hybridizing corpus-based information, with a genetic fuzzy approach and semantic similarity notion. First the paper suggests use of a genetic fuzzy approach to select an optimal combination of query terms from a pool of terms obtained using PRF-based query expansion. The query terms obtained are further ranked on the basis of semantic similarity with original query terms. The experiments were performed on CISI collection, a benchmark dataset for information retrieval. It was found that the results were better in both terms of recall and precision. The main observation is that the hybridization of various techniques of query expansion in an intelligent way allows us to incorporate the good features of all of them. As this is a preliminary attempt in this direction, there is a large scope for enhancing these techniques.

Keywords
Genetic fuzzy algorithm; information retrieval; pseudo relevance feedback; query expansion; semantic similarity

1. Introduction

This section presents an overview of information retrieval system and focuses on need for query expansion. Further, it discusses the appropriateness of evolutionary approaches and the need for incorporating semantics in the field of query expansion.

1.1. Information retrieval: an overview

The history of information retrieval (IR) parallels the development of libraries. The first civilizations had already come to the conclusion that efficient techniques should be designed to fully benefit from large document archives. Only recently, IR has radically changed with the advent of computers and digital technologies. Digital technologies provide a unified infrastructure to store, exchange and automatically process large document collections. The search for information consequently evolved from the manual examination of brief document abstracts within predefined categories to algorithms searching through the whole content of each archived document. There are a large number of applications in which information retrieval is useful. Nowadays, automatic retrieval systems are widely used in several application domains, including digital libraries, information filtering, recommender systems, media search and search engines, and

Corresponding author:
Pragati Bhatnagar, M.L. Sukhadia University, Udaipur, Rajasthan, India.
Email: pragatibhat@gmail.com

there is a constant need for improving such systems. In this context, IR is an active field of research within computer science.

One of the main tasks of IR, the so-called ad-hoc retrieval task, aims at finding the documents relevant to a submitted query. This problem is generally formalized as a ranking problem: given a query, the retrieval system should rank the documents, so that the documents more relevant to the query appear above the less relevant documents.

1.2. Information retrieval system

An information retrieval system (IRS) [1] retrieves information in response to user queries. It stores a document corpus, accepts a user query, searches the document collection and returns a ranked set of relevant documents. An IRS consists of three basic components: a documentary database, a query subsystem and a matching mechanism [1].

1.2.1. The documentary database. This component stores the document corpus. It is associated with the indexer module, which automatically generates a representation of each document by extracting the document contents. Textual document representation is typically based on index terms (single terms or phrases), which are content identifiers of document value. The system first extracts indexing terms from documents and then assigns weights to these terms using different approaches. The efficiency of such systems is based on how efficiently they solve two major problems: one is how to extract keywords precisely and other is how to decide the weight of each keyword.

1.2.2. The query subsystem. This allows the user to specify their information needs. To do this it requires a query language that allows the user to formulate a query and a procedure to pre-process the query so that it can be matched with documents in the database.

1.2.3. The matching mechanism. This evaluates the degree to which documents are relevant to user query, giving a retrieval status value for each document. The relevant documents are ranked on the basis of this value. The accuracy of an IRS is very much dependent on appropriate selection of matching mechanism.

1.2.4. Models for information retrieval systems. In order to implement an IRS several models have been proposed. Some of these models are the Boolean model, the vector space model and the probabilistic model. Vector space model is most popularly and computationally efficient model used in information retrieval system. In this model both documents and query are represented in the form of a term vector and a vector-based similarity measure such as jaccard, cosine, and Okapi, which is used for matching the query and the document.

1.3. Query expansion and evolutionary approach

1.3.1. Need for query expansion. The main objective of an information retrieval system is to return the most relevant documents corresponding to the user query. However there are many problems in developing an efficient information retrieval system. One of the major problems faced by an IRS is the word mismatch problem [2, 3]. The word entered by the user may have synonyms (different words with similar meanings) and a specific word may be polysym (same words with different meanings). Owing to the word mismatch problem, concepts may be described in different words in user queries and/or documents. According to Furnas et al. [2], people use the same term to describe an object less than 20% of the time. For example, a user may use ‘collection’ to describe a group or collection of documents collected on a specific topic while authors may use ‘corpus’ for the same concept. Evidently, the word mismatch problem, if not addressed by the IRS, would degrade the retrieval effectiveness. This problem is more severe for short queries (i.e. queries with fewer terms) than long queries because, as a query gets longer, the probability of its term co-occurrence with the terms of relevant document increases [3, 4].

Query expansion techniques help in overcoming the word mismatch problem to a certain extent. Query expansion has been a motivation for a long time for improving the retrieval efficiency of an IRS. The query expansion may be done in different ways: manual (the user chooses expansion terms), interactive (the system suggests expansion terms to the user) and automatic (the whole process is invisible to the user). Obviously interactive query expansion may be better than automatic query expansion as both user and system are involved. However in most of the cases it is not feasible to involve the user in the process of query expansion; therefore many researchers are trying to develop efficient techniques for automatic query expansion. A good survey of automatic query expansion given in Carpineto and Romano [5].

1.3.2. Sources for finding expansion terms. There can be multiple sources for selecting the query expansion terms. These sources can be grouped as follows: document corpus (for global, local, relevance feedback-based query expansion), linguistic resources (dictionary, thesaurus, Wordnet Ontology, for semantic query expansion) and world knowledge-based resources (Wikipedia). Most of the query expansion approaches use a particular type of source. Each one of them has its own nature, advantages and limitations and provides specific types of terms. However these sources, when combined properly, may significantly enhance the performance of query expansion. Recently researches have started combining these resources and developing hybridized query expansion approaches.

1.3.3. Appropriateness of evolutionary approaches for query expansion. There are strong reasons for using evolutionary techniques in the field of query expansion. Broadly query expansion can be translated to an optimization problem, where it is expected that an appropriate combination of expansion terms is selected in order to maximize the performance of an IRS. The objective function to be optimized is based on the effectiveness of the query to retrieve relevant material when presented to the IRS. Depending upon the IRS goals precision, recall or other customized performance evaluation parameter may be used. However as there are no straightforward approaches/methods to solve such types of optimization problems, evolutionary approaches can be appropriately used in such cases. As in the case of information retrieval, the search space is generally very large and multidimensional; therefore, a genetic algorithm (GA) is a suitable evolutionary approach. In some cases a multiobjective solution is required; here also a multiobjective GA can be useful. Finally a good solution requires exploration and exploitation in each direction of the search space. In such cases crossover and mutation operators work well.

1.4. Incorporating semantics in query expansion

While dealing with natural language text, it is important to consider the text at semantic level. However, owing to subjectivity in textual data and high computational cost, deep semantic analysis is neither feasible nor advisable. To incorporate the notion of semantics, external linguistic knowledge bases such as linguistic ontology can be used. Linguistic ontology provides a computationally efficient approach to find relations (synonym, hypernym, hyponym) and similarities amongst linguistic entities. Some work has been done on expanding the query using WorldNet ontology. However these methods alone are not very successful in improving the query. In this paper we suggest incorporating semantics with corpus-based query expansion. We suggest that this hybridization can be very useful and allows us to use the notion of semantics in an intelligent fashion.

2. Related work

Query expansion has been widely investigated as a method for improving the performance of IRSs. Although a lot of work has been done in this area, very limited success has been obtained. Obtaining a proper expansion of a query is still an unsolved problem and is a challenging area of research. Different researchers are coming up with different approaches to query expansion. Keeping aside the approaches for any type of query expansion, there are two important concerns in query expansion: the source for obtaining expansion terms and criteria for selecting and ranking the expansion terms.

The most popular approach for expanding the query is the corpus-based approach, where the expansion terms are selected from the document corpus itself. This type of expansion can be divided in two types: global and local query expansion. In global query expansion the entire corpus is considered for selecting the expansion terms, whereas in local query expansion the terms are selected from an initially retrieved collection of relevant documents. This collection of documents is retrieved using a matching function. The matching is generally based on lexicographic similarity measures such as cosine, jacquard or dice [1]. However recently it has been shown that a relatively new measure, the Okapi similarity measure, is giving good results in comparison to other similarity measures [6]. Once the documents have been retrieved, the user is involved in selecting documents in the collection that are actually relevant. This subset of documents acts as a source for selecting expansion terms. This type of expansion is called *relevance feedback-based query expansion*. However involving the user in the process is generally a complicated task and is not feasible. Therefore another popular type of query expansion, *pseudo relevance feedback*-based *query expansion* [7], is becoming more popular. In pseudo relevance feedback (PRF)-based query expansion, the top *n* retrieved documents are considered to be relevant and are used as a source for selecting the expansion terms. The next step in PRF-based query expansion is the selection of expansion terms. The most natural way of selecting the terms is to select the terms that are co-occurring with the query terms.

In fact idea of co-occurrence is based on the Association Hypothesis: 'If an index term is good at discriminating relevant from nonrelevant documents then any closely associated index term is likely to be good at this'. Methods based on term co-occurrence have been used since 1970s to identify the relationships that exist among terms. As early as 1969, Lesk [8] gave the idea of word associations. Lesk expanded a query by including those terms that have a similarity to a query term greater than some threshold value of the cosine coefficient. Van Rijsbergen [9] provided a theoretical basis for using co-occurrence statistics to detect the semantic similarity between terms and exploited it to expand the user's queries. The main problem with the co-occurrence approach was reported by Peat and Willett [10], who claimed that similar terms identified by co-occurrence also tend to occur very frequently in the collection and therefore these terms are not good elements to discriminate between relevant and nonrelevant documents. This is true when the co-occurrence analysis is done generally on the whole collection (global) but if we apply it only on the top ranked documents (local) discrimination does occur to a certain extent. Therefore co-occurrence-based techniques have been applied more successfully on PRF-based query expansion.

An efficient co-occurrence-based measure has been proposed for ranking the query expansion terms in PRF and co-occurrence-based query expansion [3, 4]. Information theoretic measures have been used to re-rank these terms to improve retrieval efficiency [11]. However Cao et al. [12] even questions the basic notion of the goodness of a term. They argue that goodness criteria which are based on the frequency of terms in PRF-based documents or their distribution in the corpus itself are not appropriate. The authors then propose integrating a term classification process to predict the usefulness of expansion terms.

A query can be expanded by selecting the terms from an external knowledge source such as a dictionary, thesaurus, ontology, etc. A good amount of work has been done where Wordnet Ontology [13, 14] has been used to find the relation and similarity between Linguistic entities. Based on the semantic relations (synonym, hypernym, hyponym) available in WordNet and the graphical structure of the words/concepts in Wordnet, various semantic similarity approaches have been developed. Some significant contributions led to the development of standard modules for finding semantic similarity/distance between the words. These include: node-based measures [15, 16], edge-based measures [17, 18], feature-based measures [19] and hybrid measures [20].

More recently WordNet has been used for information retrieval and query expansion. Verelas [21] described an application of WordNet-based semantic similarity measures in IR. Voorhees [22] expanded queries using a combination of synonyms, hypernyms and hyponyms manually selected from WordNet, and achieved limited improvement on short queries. Stairmand [23] used WordNet for query expansion, but he concluded that the improvement was restricted by the coverage of the WordNet and no empirical results were reported. Liu et al. [24] used WordNet for both sense disambiguation and query expansion and achieved reasonable performance improvement. However, the computational cost is high and the benefit of query expansion using only WordNet is unclear. Another important problem with such processes is the need to disambiguate the sense of the words of the query, which itself is a difficult task.

Some work has been done using GAs for information retrieval and query expansion. Most of the work has been done to tune the weights of query terms or matching functions. Pathak et al. [25] used a GA for improving the efficiency of matching function of an IR system. Horng [26] used a GA to tune the weight of retrieved query terms. The experiment was performed on a Chinese data collection. Araujo et al. [27] used a GA for query expansion based on stemming and morphological variations. Cecchini et al. [28] used a GA along with the notion of thematic context to improve query expansion. The proposed techniques place the emphasis on searching for novel material that is related to the search context. Calumby et al. [29] presented a framework for multimodal retrieval with relevance feedback based on genetic programming. Eleftherios et al. [30] presented research on a system focused on retrieving information about activities related to a simple query which is given as input. An overview of the system's design based on semantic query expansion was given along with detailed explanation of the optimization of the system's parameters through the use of a GA.

In order to improve the efficiency of GAs, several hybridizations have been suggested. Some work has been done in different areas, where some heuristics /rules can be framed and other soft computing techniques such as neural networks, PSO and fuzzy rules can by hybridized with GAs to improve performance. Schuster [31] proposed heuristics for optimal setting of the mutation probability (P_c).Fogarty [32] and Booker [33] investigated time dependencies on the mutation and crossover probabilities, respectively. Grefenstette [34] and Schaffer et al. [35] found optimal settings for all these parameters of the GA by experiment. Chaturvedi et al. [36] developed a GA approach for changing ANN weights during training by modifying GA parameters (P_c and P_m) using a fuzzy approach. Saini et al. [37] presented a GA-fuzzy-based approach for solving the Optimal Power Flow problem. The GA parameters, such as crossover and mutation probabilities, are governed by a fuzzy rule base. There is a need to evaluate the worthiness of such hybridization in the IR field also. However little work has been done in this direction. Borkar and Patil [38] presented a model of hybrid GA-particle swarm optimization for web information retrieval.

3. Background and motivation

Our approach is focused around two objectives. First we propose use of a genetic approach hybridized with fuzzy rules for improving efficiency of PRF-based query expansion. Second we suggest that semantic similarity methods can be used to filter the noisy terms.

Our approach is important as we have not found any study where GAs and notions of semantic similarity have been combined to improve the performance of PRF-based query expansion. However, we highlight the work which is similar to our own and has motivated us to use GAs for our work. Araujo and Perez Aguera [27] proposed a GA-based approach for improving query expansion with stemming terms. However, the main focus is on considering morphological variation of candidate terms and selecting the best combination of these variants. Here the candidate terms are selected semantically, whereas our work combines corpus-based expansion with a semantic similarity notion. Cecchini et al. [28] proposed an approach that describes optimization techniques based on GAs to evolve 'good query terms' in the context of a given topic. The proposed technique places emphasis on searching for novel material that is related to the search context. This work first requires a context to be created for the topics and this context is then used for adding the terms for expansion. The method can be applied only to specific topics for which contexts have been created. Therefore it cannot be used for a new topic for which a context is not available, whereas our approach is a general approach, which can be used for any query. Moreover this approach is built on the results obtained by a search engine; hence it is suitable for query expansion of web documents. Calumby et al. [29] presented a framework for multimodal retrieval with relevance feedback based on genetic programming. The authors used a supervised learning-to-rank framework; genetic programming is used for the discovery of effective combination functions of (multimodal) similarity measures using the information obtained throughout the user relevance feedback iterations. The work is based on supervised learning and is not for query expansion but for improving similarity measures. Here, instead of pseudo relevance feedback, relevance feedback is used, so the approach is not suitable for automatic query expansion. Eleftherios et al. [30] presented a system that focuses on retrieving information on activities related to a query which is given as input. The main focus is on semantic query expansion and GA has been used for optimizing systems parameters.

The work discussed above is somewhat related to us but our focus is completely different. We have tried to improve the efficiency of corpus-based query expansion (PRF-based query expansion) using a genetic fuzzy approach and by combining semantic similarity notion with the corpus. PRF-based query expansion is a corpus-based query expansion that does not require any external knowledge and no user intervention is required at any stage. It is a popularly used method of expanding the query. PRF-based query expansion methods have been well explored and have shown some promise. However, these methods have their own limitations and have a mixed chance of success. As discussed earlier, PRF-based query expansion augments the query with terms from the documents in an initially retrieved list. Candidate terms obtained are ranked on the basis of their co-occurrence value with the query terms. The top m terms are used for expansion. We observed that some terms are coming lower in rank and may not be considered for expansion if the top m terms are taken. However, if combined with some other important terms, the overall importance of the combination suggests that both of them should be considered together for expansion. This indicates that, instead of considering the ranking of terms individually, a good combination of the terms should be scored in totality. This indicates the appropriateness of GAs for exploring the search space, which is discussed below.

Consider there are n candidate terms and m terms are to be considered for expansion, where the value of m is determined empirically. In order to select an optimal combination of m terms, ${}^{n}C_{m}$ combinations are required to be checked to exhaust all possibilities. This makes the search space very large and the problem becomes NP hard. This suggests the need for evolutionary approaches such as GA to find an optimal solution. This is the case if we know the optimal total number of terms that should be used for expanding the query. However in most cases this number may not be fixed *a priori*, making query expansion more difficult to deal with. Depending on the nature of query and candidate terms, the number of terms that should be added should be allowed to vary. Most of the query expansion approaches first try to determine optimal number of terms that should be added to query and a fixed number of terms are added to each query. This is because it is difficult to deal computationally with variable length expansion. One of the main features of our approach is that this limit to the number of expansion terms need not be fixed and GA allow this number to be tuned accordingly. In our problem, although chromosomes are of fixed length, the nature and representation of chromosomes allows us to consider an appropriate subset of candidate terms (where length of subset is not a concern) for expansion. Thus GAs are found to be suitable for variable length query expansion. Therefore, in our approach, we have tried to combine the use of a GA with PRF-based query expansion. Moreover we have also hybridized the GA on the basis of fuzzy rules that have been used to improve the performance of these algorithms.

WorldNet is a well-explored linguistic ontology. Some work has been done using WorldNet for expanding the query. However, some important limitations have been reported, mainly the coverage problem and adding noisy terms. Also

these approaches are not computationally efficient. The worthiness of query expansion based on WordNet only has also been questioned in several papers. Our work fills this research gap by combining WordNet with PRF-based evolutionary query expansion and provides a computationally feasible approach to combine a corpus-based approach with semantics. On observing the terms obtained by genetic fuzzy-based query expansion, we found that some terms are very general and are not semantically related to the query terms. These terms are noisy terms and may lead to query drift. If these terms can be filtered, it will reduce the noise and hence improve the efficiency of query expansion. Thus we incorporated the notion of semantic similarity for filtering the noise.

4. Proposed approach

In the traditional approach of PRF-based query expansion (PRFBQE), the candidate terms for expansion are selected from an initially retrieved collection of documents. The main concerns in PRFBQE are the proper selection of a similarity measure for retrieving the initial set of documents and using appropriate criteria for selecting expansion terms. We used an efficient Okapi similarity measure for retrieving the initial set of documents, which is more efficient than the traditionally used cosine similarity measure. Further, we used a jaccard similarity measure for selecting the terms co-occurring with the query terms. The top n terms form a term pool of candidate terms.

We suggest the use a of GA for selecting an optimal subset of expansion terms. We call it GA-based query expansion (GABQE). Considering the problems and limitations of the GA, fuzzy rules were used for improving the performance of these algorithms. We call it genetic fuzzy-based query expansion (GAFBQE). Further, we used the semantic similarity between the original query terms and candidate expansion terms to rank the candidates. The terms at the bottom are filtered to eliminate the noise. We call this approach GA and semantic-based query expansion (GSBQE).

In order to present our approach, in Section 4.1 we discuss the construction of the term pool. In Sections 4.2–4.4 we discuss the GABQE, GAFBQE and GSBQE, respectively.

4.1. Construction of the term pool

In order to construct the term pool, retrieve the top n documents for the query using a matching function. In our problem a query is selected and its Okapi measure is used as a matching function. The Okapi measure [6] is given by following equation:

$$Okapi(Q, D_i) = \sum_{T \in Q} w \frac{(k_1 + 1)tf}{K + tf} \times \frac{(k_3 + 1)qtf}{k_3 + qtf} \tag{1}$$

where Q is the query that contains words T; k_1, b and k_3 are constant parameters ($k_1 = 1.2$, $b = 0.75$, $k_3 = 7.0$);

$$K = k_1((1 - b) + \left(b.\frac{dl}{avdl}\right); \tag{1.1}$$

tf is the term frequency of the term in document D_i; qtf is term frequency in query Q;

$$w = \log \frac{(N - n + 0.5)}{(n + 0.5)}; \tag{1.2}$$

N is number of documents; n is number of documents containing the term; and dl and $avdl$ are document length and average document length.

All documents and sorted on the basis of Okapi measure. All the unique terms of the top N documents are selected and are ranked on the basis of their co-occurrence with query terms. The top m terms co-occurring with original query terms are selected as candidate terms for expansion.

Now the basic question with this approach is how to select co-occurring terms, as terms can be selected in a number of ways. Some standard measures have been suggested to select co-occurring terms. For our experiments, we have used the well-known jaccard coefficient [1] as a co-occurrence measure, which is given as:

$$jaccard\_co(t_i, t_j) = \frac{d_{ij}}{d_i + d_j - d_{ij}} \tag{2}$$

where t_i and t_j are the terms for which co-occurrence is to be calculated, d_i and d_j are the number of documents in which terms occur, respectively and d_{ij} is the number of documents in which t_i and t_j co-occur.

We can apply these coefficients to measure the similarity between the query terms and terms in the documents. However there is a danger in adding these terms directly to the query. The candidate terms selected for expansion could co-occur with the original query terms in the documents (top n relevant) by chance. The higher its degree in the whole corpus, the more likely it is that candidate term co-occurs with query terms by chance. Keeping this factor in mind, the inverse document frequency of a term can be used along with the above-discussed similarity measures to scale down the effect of the chance factor. Incorporating inverse document frequency and applying normalization, the degree of co-occurrence of a candidate term with a query term can be defined as follows [4]:

$$co\_degree(c, t_i) = log_{10}(co(c, t_i) + 1) * (idf(c)/log_{10}(D)) \tag{3}$$

$$idf(c) = log_{10}(N/N_c) \tag{3.1}$$

where N = number of documents in the corpus; D = number of top ranked documents used; c = candidate term listed for query expansion; t_i = ith term of the query; N_c = number of documents in the corpus that contain c; and $co(c,t_i)$ = number of co-occurrences between c and t_i in the top-ranked documents, that is, $jaccard\_co(c_i, t_j)$.

The above formula can be used for finding similarity of a term c with the individual query term. To obtain a value measuring how good c is for whole query Q, we need to combine degrees of co-occurrence of c with all individual original query terms. We use equation (4) for finding the similarity between the candidate term c and the query Q:

$$SuitabiltyforQ = f(c, Q) = \prod_{t_i in Q} (\delta + co\_degree(c, t_i))^{idf(t_i)} \tag{4}$$

The above equation provides a suitability score for ranking the terms co-occurring with the entire query. The terms are ranked on the basis of similarity value obtained and the top m terms from the term pool.

4.2. Genetic algorithm-*based* query expansions

We have discussed the approach for developing the term pool. The term pool contains the good candidate terms for expanding the query. Now we have to select an optimal combination of a subset of these terms, which are cohesive among themselves and are better suited for query expansion. We discuss the use of a GA for selecting the expansion terms. In order to apply the GA we require a proper fitness function. The performance of the GA is very much dependent on proper representation of chromosomes, and the proper selection and tuning of crossover and mutation operators. In this section we discuss these issues.

4.2.1. Representation of chromosomes. As our population consists of a combination of candidate terms, we used a chromosome representation where each gene represents a specific candidate term. We took binary representation for chromosomes. We used 50 candidate terms in all; therefore the length of chromosomes is 50. Each location is representative of a specific term. A value of 1 at a location indicates that the term should be considered for expansion, whereas 0 indicates otherwise.

4.2.2. Fitness function. The fitness function is based on the suitability or goodness of the query in retrieving the relevant documents, which is measured by recall, precision or f measure (harmonic mean of recall precision). Recall is given by:

$$Recall = \frac{|R_a|}{|R|} \tag{5}$$

where R_a is the set of relevant documents retrieved and R is the set of all relevant documents.

$$Precision = \frac{|R_a|}{|P|} \tag{6}$$

where P is the set of top n retrieved documents.

1. Select the query.
2. Find similarity measure of all documents with query.
3. Sort documents according to their similarity measure.
4. Find all unique terms of top *n* retrieved documents, giving a term set *T*.
5. Calculate co-occurrence between each query term and each term of set *T* using jaccard similarity measure (eqn (2)).
6. Find similarity of whole query with each term of *T* giving suitability score (eqn (4)) to each term of *T*.
7. Rank the terms of *T* on the basis of similarity obtained in step 6.
8. Form a term pool of candidate expansion terms containing top *m* terms obtained from step 6.
9. Perform step 9a–b for applying GA
   9a. Generate initial population randomly from the term pool.
   9b. Repeat steps 9bi–9bii until the population converges or for maximum number of generations.
      9bi. Calculate fitness value for each population members (a–b):
         a. Expand the original query by adding terms of the individual population member.
         b. Retrieve the initial set of documents.
         c. Calculate the fitness of the expanded query using recall-based measure (eqn (5)).
      9bii. Form a new population using selection, crossover and mutation operation.
10. Return the combination of terms obtained in final generation, with maximum fitness as final set of expansion terms.

Figure 1. Algorithm developed for selecting expansion terms.

4.2.3. GA operators. Selection, crossover and mutation are the GA operators that are applied to the chromosomes. Selection embodies the principle of survival of the fittest. Crossover is the genetic operator that combines two selected chromosomes to form a new chromosome. Chromosomes having higher fitness have a higher probability of participating in the crossover operation. Using single-point crossover, a locus position is selected within two parent chromosomes and the genes are swapped from that position to the end of parent. Mutation is another genetic operator that works on specific genes of a single chromosome. In our case mutation is an inversion operator, inverting the availability/unavailability of term for expansion.

4.2.4. Algorithm for genetic algorithm-based query expansion. Once the tem pool is constructed for selecting the candidate terms and the GA has been properly defined with a suitable representation of chromosomes, appropriate fitness function and genetic operators, we present an algorithm for GABQE in Figure 1.

5. Genetic fuzzy-based query expansion

When using a GA, it is important to tune the proper crossover rate and mutation rate. It has been observed that after few generations, the fitness value of each chromosome becomes almost equal to that other chromosomes from the same population. The effect of crossover and mutation beyond this stage becomes insignificant owing to very small variations in the chromosomes in a particular population. This requires changing the crossover and mutation rate based on population statistics. This tuning has to be done empirically. It is difficult to tune these parameters properly. On the basis of intuition and experience, fuzzy rules have been framed to improve the performance of the GA [37]. The rules are applied when it is observed that the GA is not performing well. For measuring the performance of the GA, one has to observe the results of the GA for a number of successive generations. The following rules allow us to judge that the GA is not performing well:

- best fitness remains low for successive generations;
- fitness of all chromosomes becomes almost the same and remains the same for many generations;
- best fitness remains same over many generations.

The probable reasons behind the above problems associated with the GA may be that the population has stuck in local maxima or the region of the search space being covered does not contain the solution. In such cases, the use of crossover or mutation becomes insignificant. Thus the crossover and mutation rates have to be varied significantly and the GA is able to give an optimal solution only after proper tuning. For example, if best fitness is low, *crossover* rate needs to be increased and mutation needs to be reduced. Similarly other heuristics have to be explored to properly explore the search space and get an optimal solution.

Table 1. Membership functions and range of variables for linguistic variables.

Linguistic variable	Nature of fuzzy set	Range of variable		
		Low	Medium	High
Crossover probability (P_c)	Triangular	0.5, 0.7	0.6, 0.8	0.7, 0.95
Mutation Probability (P_m)	Triangular	0.005, 0.02	0.01, 0.03	0.02, 0.1
Best Fitness (BF)	Triangular	0, 0.7	0.5, 0.9	0.7, 1.0
Number of generations for unchanged BF (UN)	Triangular	0, 6.0	3.0, 9.0	6.0, 12.0
Variance of fitness (VF)	Triangular	0, 0.12	0.1, 0.14	0.12, 0.2

Fuzzy rules have been used to tune the crossover and mutation parameters of GA experiments. For this the fuzzy linguistic variables used are BF (best fitness), UN (number of generations fitness remains unchanged) and VF (variance of fitness). Fuzzy sets for these values and the fuzzy variables are defined below.

The following fuzzy rules have been used to tune the values of crossover and mutation parameters:

(1) For controlling P_c
  (1.1) If BF is LOW then P_c is HIGH.
  (1.2) If BF is MEDIUM or HIGH and UN is LOW then P_c is HIGH.
  (1.3) If BF is MEDIUM or HIGH and UN is MEDIUM then P_c is MEDIUM.
  (1.4) If UN is HIGH and VF is LOW or MEDIUM then P_c is LOW.
  (1.5) If UN is HIGH and VF is HIGH then P_c is MEDIUM.

(2) For controlling P_m
  (2.1) If BF is LOW then P_m is LOW.
  (2.2) If BF is MEDIUM or HIGH and UN is LOW then P_m is LOW.
  (2.3) If BF is MEDIUM or HIGH and UN is MEDIUM then P_m is MEDIUM.
  (2.4) If UN is HIGH and VF is LOW then P_m is HIGH.
  (2.5) If UN is HIGH and VF is MEDIUM or HIGH then P_m is LOW.

The membership functions controlling the linguistic variables are given in Table 1. After incorporating fuzzy rules we come up with an enhanced approach for GABQE. We call it genetic fuzzy-based query expansion.

6. Genetic algorithm and semantic-based query expansion

Once we obtained the terms using GAFBQE, we observed that some terms are noisy in the sense that they are very general and not related to query. In such cases these terms may retrieve some irrelevant documents. Thus it is important to filter these terms. In order to obtain a refined set of the terms, we tried using some background knowledge in the form of Linguistic ontology WordNet. The idea was that, if the term has some semantic relation to the query term, it is more appropriate for expansion. However we could not find an example of WordNet relations (such as synonym, hypernym, hyponym) holding between query and expansion terms. We observed that the some expansion terms were semantically similar to query terms and were seemingly more important for query expansion. As discussed in Section 2, a number of semantic similarity measures have been developed to find a quantitative measure of similarity between two words. These measures are based on the notion of content of information shared between two words and their shortest path distance within Word Net Ontology. We used the Wu Palmer semantic similarity measure in our work and found that the results are promising.

Overall we have tried to improve the performance of genetic-based query expansion by hybridizing corpus-based information with semantic information and we name this approach GSBQE. Our approach of GSBQE takes as input the expansion terms obtained after GAFBQE. We give a new formula for finding the suitability of candidate expansion terms for query expansion, on a pattern similar to co-occurrence formula (4). However, this formula is based on semantic similarity rather than co-occurrence-based similarity. We suggest eqn (7) for finding semantic similarity between the candidate expansion term and the query:

$$SemanticSimilarityforQ = f(c, Q) = \sum_{t_i inQ} semanticsimilarity(c, t_i) \tag{7}$$

Table 2. Selected queries.

Query no. 2	How can actually pertinent data, as opposed to references or entire articles themselves, be retrieved automatically in response to information requests?
Query no. 12	Give methods for high-speed publication, printing and distribution of scientific journals.
Query no. 28	Computerized information systems in fields related to chemistry.
Query no. 34	Methods of coding used in computerized index systems

Table 3. Results showing average precision and recall for various methods.

Query no.	Original query		PRFBQE		GABQE		GAFBQE		GSBQE	
	Rec	Prec	Rec	Prec	Rec	Prec	Rec	Pre	Rec	Prec
2	0.0769	0.02	0.1583	0.0411	0.2038	0.0533	0.2308	0.0529	**0.34**	**0.0884**
12	0.4615	0.0600	0.234	0.0304	00.521	0.0677	**0.581**	**0.075**	0.59	0.0708
28	0.2833	0.1700	0.2333	0.1399	0.425	0.255	**0.512**	**0.307**	0.50	0.296
34	0.3421	0.1100	0.2895	0.1011	00.473	0.1801	0.51	0.1938	**0.53**	**0.2014**
All queries (average)	0.212	0.08	0.47	0.16	0.59	0.2	**0.63**	**0.23**	0.51	0.19

where Q is the query, c is the candidate term and t_i is the ith term of the query.

Semantic similarity represents semantic similarity between two terms (query term and candidate expansion term). This value can be calculated using a semantic similarity module/approach such as Resnik, WuPalmar, Jiang or Korth [14–17], which takes two words/concept as input and returns semantic similarity between these two terms. *SemanticSimilarityforQ* provides a semantic similarity value for entire query and candidate term. The candidate expansion terms are sorted on the basis of their suitability for the query. The top n terms or all the terms above a certain threshold are finally selected as expansion terms. This results in filtering out the terms with low semantic similarity and improving the rank of the terms with high semantic similarity.

7. Experiments and result

We performed the experiments on the CISI dataset. This dataset provides a benchmark for testing the efficiency of an IRS. The CISI data consist of 1460 abstracts from information retrieval papers and 112 queries. A comparison was made for unexpanded query, PRF-based query expansion, GABQE, GAFBQE and GSBQE. Recall and precision measures were taken for measuring the performance of the system. For PRF-based query expansion and all other experiments, the Okapi similarity measure was used for initial ranking of the documents. The performances of various approaches of query expansion for some selected queries and on average are presented as follows. The selected queries are shown in Table 2 and the result is presented in Table 3.

For PRF-based query expansion, candidate terms were obtained from the top 10 initially retrieved documents. The terms were ranked on the basis of their co-occurrence-based similarity with the query using eqn (4). The top 10 terms were used for expanding the query. The values of the top n (10) number of documents and number of terms for expansion (10) were selected by referring to other works and extensive experiments on the dataset. The results showed a substantial improvement in average recall and precision of queries. The average result is shown in Table 3, row 5. However a large variation in performance of individual queries was observed. In spite of increase in performance of most of the queries, for some queries performance was degrading.

As discussed in Section 3, the objective of applying GABQE algorithm was to obtain an optimal combination of terms, out of the top m (50) candidate terms obtained from the PRF-based query expansion. For GABQE the GA was executed for 50 generations and the population size was 40. Crossover rate and mutation rate were set empirically after 20 runs of GA. Finally the crossover rate was taken as 0.7 and the mutation rate as 0.03. The recall value was considered as the fitness function. Figures 2 and 3 show average recall and average precision of all the queries generation-wise. It can be observed that average fitness (recall) is increasing and slowly reaches convergence. Figures 4 and 5 show generation-wise average recall and precision of query no. 2. From the figures it can be seen that, after applying the GA, recall increases slowly and converges ultimately. This shows the improvement in retrieval of documents by expanding queries using GA.

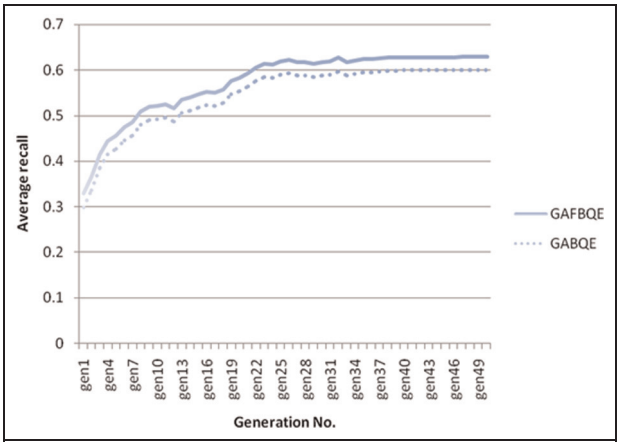

Figure 2. Average recall for all queries vs generation number.

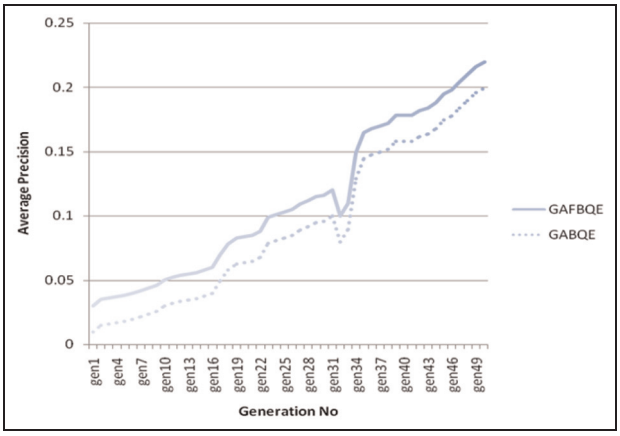

Figure 3. Average precision of all queries vs generation number.

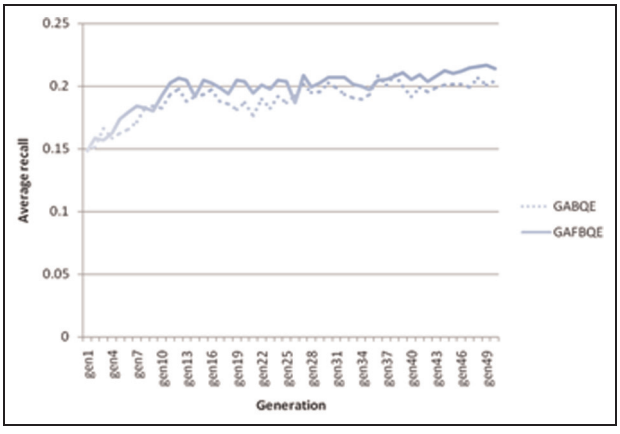

Figure 4. Average recall of query 2 vs generation number.

In case of expanding the query with PRFBQE (without GA), recall is degraded by small amount in some cases, whereas it is increasing in all most all cases of GABQE (see query nos 12, 28, 34 Table 3). As candidate expansion terms are the same in both the cases, it is observed that the GA enhances the performance as it allows selecting a better

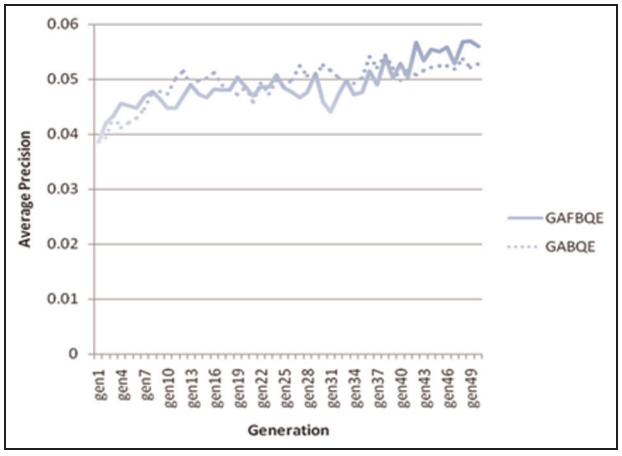

Figure 5. Average precision of query 2 vs generation number.

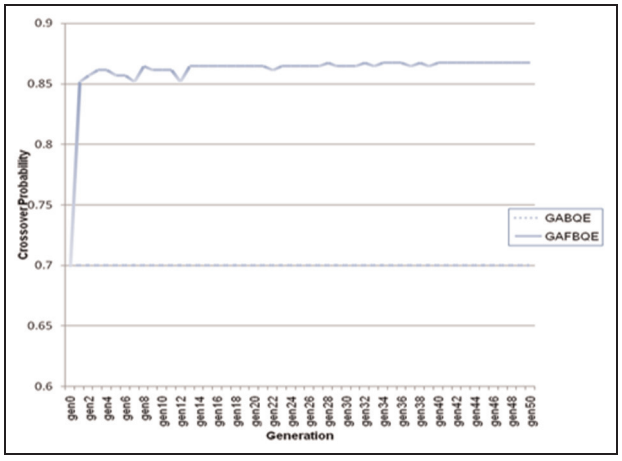

Figure 6. Crossover probability (P_c) variation for GABQE and GAFBQE method.

and cohesive combination of expansion terms. Therefore, application of GA helps in expanding the query in such a manner that it improves efficiency of information retrieval.

The GA was combined with fuzzy rules to perform GAFBQE. The improvement in performance of GAFBQE over GABQE can be observed in Figs 2–5. Fuzzy rules were used to control the crossover (P_c) and mutation rate (P_m), resulting in improved performance of GA. The variation in P_c and P_m with generation is shown in Figs 6 and 7, respectively. It can be observed that P_c starts changing from generation 1 and becomes constant (0.87) after generation 13. Similarly P_m starts changing from generation 1 and becomes constant (0.01) after generation 13.

Finally the experiments were performed for GSBQE. This approach allows us to combine the notion of semantics with corpus-based query expansion. In this method we used equation 7 to find similarity between the terms obtained for expansion (using GAFBQE) and the query. The expansion terms were ranked on the basis of this similarity and terms with zero similarity were filtered.

The final results of all methods on average and for selected queries are shown in Table 3. The best results are shown in bold. In order to analyse the query wise result, we observed the expansion terms obtained for individual queries. The expansion terms obtained are presented in Table 4. It can be observed that in most of the cases the query terms obtained seem to be more appropriate as we move from left to right in Table 4. For example the terms obtained – citation, journal, semantic abstract (query no. 2, GSBQE, last column) – are more appropriate than the terms obtained for other methods. Correspondingly the result obtained is also best amongst all methods (Table 3). For many queries GSBQE improves performance over GAFBQE. However in many cases GSBQE is unable to capture specific relations and filters important terms, thus reducing the performance. For instance for query 28, specific terms such as asca (advance scale conditioning

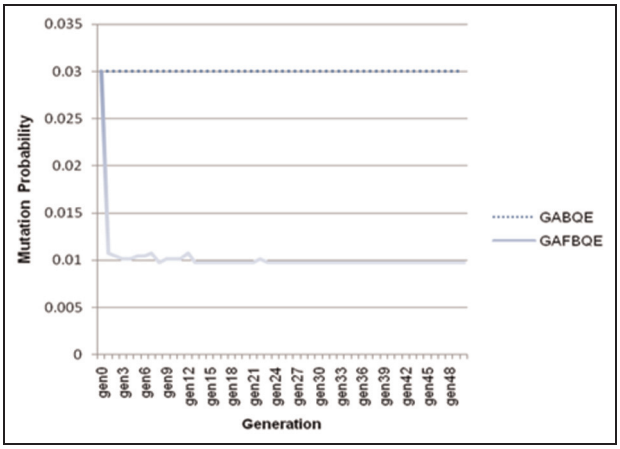

Figure 7. Mutation probability (P_m) variation for GABQE and GAFBQE method.

Table 4. Expansion terms obtained for different approaches for some selected queries.

Query no.	Original query terms	Expansion terms obtained with PRFBQE	Expansion terms obtained with GAFBQE	Expansion terms obtained with GSBQE
2	actually articles automatically data entire information opposed pertinent references requests response retrieved	list, file, journals, abstract, journal, exact, designed, source, cent, sources	syntactic abstract semantic document report list journal citation exact design source	list citation journal syntactic abstract source semantic
12	distribution high journals methods printing publication scientific speed	journal citation citations papers including articles coverage total abstracting primary	discovery citation articles coverage papers total abstracting primary author multiple work past able references year documentation source national inversion standard	references work source discovery papers citation documentation articles total standard coverage
28	chemistry computerized fields information related systems	title, considerably, similarities, estimating, synthesis, mathematics, english, citation, cited, alternative	easily, included, asca, chemical title, sdi, estimating, alternative, file, compounds title synthesis classification	title synthesis classification
34	coding computerized index methods system used	base brief accrue Rules derived conventional codes order developed compounds	developed brief based followed compounds generate available chemical entries efficiency storage searching machine	entries storage machine brief compound chemical

agent) and sdi (*SDI* Biomed provides the highest quality of laboratory testing products for use with laboratory *chemistry* analysers) that are obtained using GAFBQE, which are related to chemistry, are filtered in GSBQE. This could be included if some domain-specific ontology is used.

8. Conclusion

PRF-based query expansion is a well-studied research field in the area of information retrieval. However it has not been explored much from the point of view of using evolutionary techniques. The paper provides an approach for the use of evolutionary techniques and the semantic similarity notion for improving PRF-based query expansion and provides an

insight to explore this area of research. Specifically the paper suggests use of GABQE, GAFBQE and GSBQE in order to improve the retrieval efficiency of an information retrieval system.

The experiments were performed on the standard CISI collection, a standard dataset for IR experiments. The comparison of the results was done on the basis of recall and precision. The average results have been reported and efforts have been made to perform query-wise analysis of results by observing the expansion terms obtained. The results are promising. It was observed that GABQE provides a more cohesive and better selection of expansion terms. Fuzzy rules in GAFBQE are able to improve the performance of GA. In GSBQE semantic similarity was used to select those terms which are semantically similar to the query term.

The paper hybridizes query expansion in terms of various sources (corpus-based and Word net-based) as well as various approaches (PRF-based, evolutionary and semantic) to improve the performance of query expansion. This is a preliminary attempt at such hybridization and the results are promising. In future, there is a large scope for using bagging and boosting and combining the various results in an intelligent fashion. This may result in substantial improvement in this field.

Funding

This research received no specific grant from any funding agency in the public, commercial or not-for-profit sectors.

References

[1] Grossman D and Frieder O. *Information retreival: Algorithm and heuristic*, 2nd edn. New York: Springer,2004.
[2] Furnas GW, Landauer TK, Gomez LM and Dumais ST. The vocabulary problem in human–system communication. *Communications of the ACM* 1987; 964–971.
[3] Xu J. Solving the word mismatch problem through text analysis. PhD Thesis, University of Massachusetts, Department of Computer Science, Amherst, MA, 1997.
[4] Xu J and Croft WB. Improving the effectiveness of information retrieval with local context analysis. *ACM Transactions on Information System* 2000; 18(1): 79–112.
[5] Carpineto C and Romano G. A survey of automatic query expansion in information retrieval. *ACM Computing Survey* 2012; 44(1): 1–50.
[6] Robertson SE, Walker S et al. OKAPI at TREC-3. In: *The third text retrieval conference (TREC-3)*, NIST, 1995.
[7] Robertson SE and Walker S. Microsoft Cambridge at TREC-9, filtering track. In: *Proceedings of text retrieval conference (TREC)*, 2000, pp. 361–368.
[8] Lesk ME. Word-word associations in document retrieval systems. *American Documentation* 1969; 2: 27–38.
[9] Van Rijsbergen CJ. A theoretical basis for the use of co-occurrence data in information retrieval. *Journal of Documentation* 1977; 33: 106–119.
[10] Peat HJ and Willett P. The limitations of term co-occurrence data for query expansion in document retrieval systems. *Journal of American Society for Information Science* 1991; 42(5): 378–383.
[11] Hazra I and Aditi S. Selecting effective expansion terms for better information retrieval. *International Journal of Computer Science & Applications* 2010; 7(2): 52–64.
[12] Cao G, Nie JY, Gao JF and Robertson S. Selecting good expansion terms for pseudo-relevance feedback. In: *Proceedings of 31st annual international ACM SIGIR conference on research and development in information retrieval*, 2008, pp. 243–250.
[13] Mandala TR and Hozumi T. *The use of WordNet in natural language processing systems*, Montreal, 1988, pp. 469–477.
[14] WordNet – A lexical database for the English Language, http://wordnet.princeton.edu
[15] Resnik P. Using information content to evaluate semantic similarity. *InL Proceedings of 14th international joint conference on artificial intelligence*, Montreal 1995: 448-453.
[16] Lin D. An information-theoretic definition of similarity. In: *Proceedings of 15th international conference on machine learning*, 1998.
[17] Leacock C and Chodorow M. *Combining local context and WordNet similarity for word sense identification in WordNet: An electronic lexical database*. Cambridge, MA: MIT Press, 1998, pp. 265–283.
[18] Wu Z and Palmer M. Verb semantics and lexical selection. In: *Annual meeting of the Associations for Computational Linguistics (ACL'94)*, Las Cruces, NM, 1994, pp. 133–138.
[19] Tverskey. Features of similarity. *Psychological Review* 1977; 84(4): 327–352.
[20] Jiang J and Conarth D. Semantic similarity based on corpus statistics and lexical taxonomy. In: *International conference on research in computational linguistics*,Taiwan, 1998.
[21] Verelas Voutsakis E and Raftopoulou P. Semantic similarity methods in WordNet and their application to IR on the Web. In: *Web information and data management*. ACM Press: New York, 2005, pp. 10–16.
[22] Voorhees EM. Query expansion using lexical semantic relations. In: *Proceedings of the ACM SIGIR conference on research and development in information retrieval*, NY, 1994, pp. 61–69.

[23] Stairmand MA. Textual context analysis for information retrieval. In: *Proceedings of the ACM SIGIR conference on research and development in information retrieval*, USA, 1997, pp. 140–147.
[24] Liu S, Liu F, Yu C and Meng W. An effective approach to document retrieval via utilizing WorldNet and recognizing phrases. In: *Proceedings of the ACM SIGIR conference on research and development in information retrieval*, 2004.
[25] Pathak P, Gordon M and Fan W. Effective information retrieval using genetic algorithm based matching functions adaption. In: *Proceedings of 33 Hawai international conference on science (HICS)*, Hawaii, 2000.
[26] Horng J and Yeh C. Applying genetic algorithms to query optimization in document retrieval. *Information Processing and Management* 2000; 36: 737–759.
[27] Araujo L and Perez Aguera J. Improving query expansion with stemming terms: A new genetic algorithm approach. In: *Proceeding of 8th European conference on evolutionary computation in combinatorial explosion*. Berlin: Springer, 2008, pp. 182–193.
[28] Cecchini RL, Lorenzetti CM, Maguitman AG and Brignole NB. Using genetic algorithms to evolve a population of topical queries. *Information Processing & Management* 2008; 44: 1863–1878.
[29] Calumby RT, R. Torres RD and Gonçalves MA. Multimodal retrieval with relevance feedback based on genetic programming. In: *Multimedia Tools and Applications*. Berlin: Springer Science and Business Media, 2012.
[30] Eleftherios K, Fukazawa Y and Ota J. Genetically optimizing query expansion for retrieving activities from the Web. In: *Proceedings of 2nd ACM international conference on web intelligence, mining and semantics*, Romania, 2012.
[31] Schuster P. Effect of finite population size and other stochastic phenomenon in molecular evolution. In: *Complex system operational approaches, neurobiology, physics and computers*. Berlin: Springer, 1985.
[32] Fogarty TC. Varying the probability of mutation in the genetic algorithm. In: *Proceedings of 3rd International Conference in Genetic Algorithms and Applications*, Arlington, VA, 1981, pp. 104–109.
[33] Booker L. *Improving search in genetic algorithms, genetic algorithms and simulated annealing*. London: Pitman, 1987.
[34] Grefenstette JJ. Optimization of control parameters for genetic algorithms. *IEEE Transactions on Systems, Man and Cybernetics* 1981; SMC-16(1): 122–128.
[35] Schaffer JD, Caruna RA, Eshelman IJ and Das R. A study of control parameters affecting online performance of genetic algorithms for function optimization. In: *Proceedings of 3rd International Conference on Genetic Algorithms and Applications*, Arlington, VA, 1981, pp. 51–60.
[36] Chaturvedi DK, Kumar R, Mohan M and Kalra PK. Artificial neural network learning using improved genetic algorithm. *Institution of Engineers (India) Journal* 2001; 82: 23–27.
[37] Saini A, Chaturvedi DK and Saxena AK. Optimal power flow solution: A GA-fuzzy system approach. *International Journal of Emerging Electric Power Systems* 2006; 5(2).
[38] Borkar Priya I and Patil Leena H. Web information retrieval using genetic algorithm-particle swarm optimization. *International Journal of Future Computer and Communication* 2013; 2(6).