Journal of American Science on Information and Technology

On using genetic algorithms for multimodal relevance optimisation in information retrieval

M. Boughanem [(1)] C. Chrisment [(1)] L. Tamine [(2)]

(1) IRIT SIG Université de Toulouse III, 118 Route de Narbonne, 31062 Toulouse, France

(2) GRIMM/ISYCOM Université de Toulouse II, 5 Allées A. Machado, 31058 Toulouse Cedex, France

bougha@irit.fr, *chrisme@irit.fr, tamine@univ-tlse2.fr*

Abstract

This paper presents a genetic relevance optimisation process performed in an information retrieval system. The process uses genetic techniques for solving multimodal problems (niching) and query reformulation techniques commonly used in information retrieval. The niching technique allows the process to reach different relevance regions of the document space. Query reformulation techniques represent domain knowledge integrated in the genetic operators structure in order to improve the convergence conditions of the algorithm. Experimental analysis performed using a TREC sub-collection validates our approach.

KEY WORDS : Information retrieval , multiple query evaluation, genetic algorithm, niching

1. Introduction

An information retrieval system (IRS) is devoted to finding relevant documents according to a user's need for information. This main function implies the cooperation of algorithms and models for processing, storage and retrieval. Our focus is strictly on the retrieval process of information retrieval in response to user queries.

Retrieval strategies developed in IR can be classified into two mains categories. The first one covers research on the definition of theoritical models for both documents and query representations and relevance system measures. The most common models developed are the vectorial (Salton, 1968), the probabilistic (Robertson & Sparck Jones, 1976) and the latent semantic indexing model (Dumais, 1994). In the second category of research, authors attempt to improve the results of a basic retrieval model using various strategies and utilities based on very different mathematical constructs :the Bayesian model (Turtle & Croft, 1991), the connectionist (Kwok, 1995) (Boughanem, 1997). In this context, the investigations are mainly concerned with query reformulation and multiple query evaluation. In theory, query reformulation is based on automatically changing the set of query terms and weights associated to these terms, according to statistical measures of their context meaning or some information extracted from documents retrieved and judged during the initial search (Rocchio, 1971; Robertson & Walker 1997; Harman, 1992; Haines & Croft, 1993). The multiple query evaluation has been motivated by the observation that retrieval effectiveness is often significantly improved by using a number of different retrieval algorithms (Katzer & al, 1992; McGill & al, 1979; Lee, 1997). This is because different retrieval algorithms emphasize different document and query features when measuring the relevance and therefore retrieve different sets of documents. Since different algorithms can retrieve documents with different descriptors, the overall performance of the combined algorithm may be higher.

This is the global context of our work. More precisely, we explore the basic assumption of multiple query evaluation techniques which is namely the location of multiple relevance regions in the document space. For this reason, we characterise the relevance optimisation problem as multimodal and thus propose the use of suitable genetic techniques to handle the process of information retrieval.

Genetic algorithms (Holland, 1962) constitute an interesting category of modern heuristic search. Based on the powerful principle of survival of the fittest, genetic algorithms (Gas) model natural phenomena of genetic inheritance and the Darwinian strife of survival.

In comparison to other works which propose genetic methods for multiple query evaluation (Gordon, 1988; Yang & Korfhage, 1993; Horng & Yeh, 2000), our approach is characterised by two main features:

- the exploitation of niching techniques (Goldberg, 1989) adapted to multimodal problems in order to recall relevant documents with different descriptors,
- the integration of domain knowledge in the genetic operators structure.

This paper is organised as follows. In section 2, an overview of multiple query evaluation strategies is presented. This is followed in section 3 by a general description of genetic techniques for solving multimodal problems. Section 4 outlines our genetic approach for relevance optimisation. Finally, we discuss in the last section the experimental results obtained using Mercure IRS (Boughanem, 1997) on the AP88 test collection.

2. Multiple query evaluation

2.1. General overview

The idea of combining multiple representations of either queries or texts and also using different retrieval techniques in order to improve retrieval performance has been suggested and discussed in literature under the name of data fusion.

Several works in this general area give theoretical rationales for combination techniques. The most common comes from the observation that different representations of the same query retrieve different documents (both relevant and non relevant) (Katzer & al, 1982). This may be due to the fact that the process of representation is so uncertain that any one representation captures only a part of the user's need. Thus, the combination of multiple representations will address different aspects of the user need and then retrieve more relevant documents.

Robertson (1977) also gave an interesting analysis which suggests that each representation of a query is a source of evidence and could be used to improve prediction of probability of relevance.

McGill & al (1979) and Katzer & al (1982) found that different query formulations generated different documents. However, they noticed that there was little overlap in the documents retrieved.

Turtle & Croft (1991) proposed an inference network-based retrieval model which combines different document representations and various query formulations in a probabilistic framework. They demonstrated that combining the retrieval results of natural language and boolean query formulations improves the effectiveness of IR. Belkin & al (1993) investigated the effect of progressively cumulating the evidence of various independently generated query representations of one type in a probabilistic-inference network retrieval system. Experiments carried out on a TREC test collection showed that an appropriate combination of different boolean query formulations has a positive effect upon retrieval performance. Lee (1997) analysed the research results obtained in the data fusion theory literature and suggested a new rationale for evidence combination of different runs. They investigated different combining methods and showed that using rank order of the retrieved documents gives better retrieval effectiveness than using similarity if the runs in the combination generate different rank-similarity curves.

2.2. Multiple query evaluation based on genetic combination

Genetic techniques combining retrieval results have been proposed by several authors.

Gordon (1988) adopted a GA to derive better descriptions of documents. Each document is assigned N descriptions represented by a set of indexing terms. Genetic operators and relevance judgement are applied to the descriptions in order to build the best document descriptions. The author showed that the GA produces better document descriptions than those generated by the probabilistic model.

Redescription improved the relative density of co-relevant documents by 39,74% after twenty generations and 56,61% after forty generations.

Yang & Korfhage (1993) proposed a GA for query optimisation by reweighting the query term indexing without query expansion . They used a selection operator based on a stochastic sample, a blind crossover at two crossing points, and a classical mutation to renew the population of queries. The experiments showed that the queries converge to their relevant documents after six generations.

Kraft & al (1995) apply GA programming in order to improve the weighted boolean query formulations. Their first experiments showed that the GA programming is a viable method for deriving good queries.

Horng & Yeh (2000) propose a novel approach to automatically retrieve keywords and then uses genetic techniques to tune the keywords weights. The effectiveness of the approach is demonstrated by comparing the results obtained to those using a PAT-tree based approach.

In this work, our goal is to develop a specific genetic model for multiple query evaluation. As the main aim of this technique is dispersion of the relevance regions in the document space, we propose the use of a suitable genetic technique for solving multimodal problems, ie niching (Goldberg, 1989) (Mahfoud, 1995). Rather than processing a traditional GA which generates a unique optimal query corresponding to similar descriptors of assumed relevant documents, the integration of the niching method will tune the genetic exploration in the direction of the multiple relevant documents. Furthermore, we propose the use of enhanced genetic operators exploiting knowledge related to relevance feedback techniques.

3. Multimodal optimisation using genetic techniques

The goal of a multimodal optimisation process is to find multiple and diverse optima across the search space of a given problem. Convergence may occur to some degree within local regions but diversity must prevail across the most prominent regions. It is well known, however, in GA theory, that the selection pressure causes the phenomena of *genetic drift* which corresponds to convergence in local regions. Informally, the term *selective pressure* is widely used to characterise the strong respectively weaker emphasis of selection on the best individuals. Thus, various techniques for reducing the selection pressure have been proposed (Baker, 1985) (Goldberg, 1989) (Fonseca & Fleming, 1995) but are not overly selective as they generally geographically close solutions to be reachable.

Dejong (1975) has proposed another technique based on an iterative execution of the GA. Using the assumption that the probabilities of reaching the multiple optima are equal, the number of executions required is computed using the following formula :

$$p^*\sum_{i=1}^{p}\frac{1}{i}\cong p^*(\alpha+\log p)$$

p : number of optima

α = 0.577, Euler constant

However, this method gives bad results in real life applications (Talbi, 1999).

In this study, we restrict our attention to niching techniques. Various other techniques for promoting genetic diversity are presented in (Mahfoud, 1995) (Horn, 1997). A niching method is based on the formation of subpopulations (subsets of the whole population) which explore different regions of the search space using subsets of individuals. Each subset called *niche* is exploited by the optimisation process in order to explore the corresponding search direction. In fact, according to GA theory (Goldberg, 1989), a set of genetic individuals traduce a *schema* which represents a part of the solution space. The most common niching approaches are presented in the following sections.

3.1. Sequential niching

This approach is based on a sequential location of multiple niches using an iterative run of a traditional GA. Beasly & al (1993) present a sophisticated strategy where at the end of each run, their algorithm depresses the fitness function at all points within a certain radius of the fittest solutions. This transformation encourages the optimisation process to explore other areas of the search space.

3.2. Ecological niching

This approach is based on the creation and exploitation of multiple environments of evolution. The basic theory of the ecological niching approach proposes a simultaneous coevolution of subpopulations of individuals which are implicitly able to use food resources. Individuals unable to properly use resources will die. Thus, the environment varies over time in its distribution of food resources, with individuals that are geographically close tending to experience the same environment (Mahfoud, 1995). The sharing (Goldberg & Richardson, 1987) and clearing techniques (Petrowski, 1997) presented below are based on this ecological inspiration.

3.2.1. Sharing technique

Goldberg & Richardson (1987) presented an implementation of the concept known as the *sharing method.* In the context of their study, each individual in a niche can consume a fraction of the available resources : the greater the population size of the niche, the smaller the fraction. This leads towards a steady state in which subpopulation size are proportional to the amount of the corresponding available resources. The general formula of the sharing fitness function is the following (Goldberg & Richardson, 1987):

$$f'(x)=\frac{f(x)}{\sum_{y\in Pop} sh(dist(x,y))}$$

x,y : individuals of the population Pop

f(x) : initial fitness function

sh(dist(x,y)) : sharing function

The sharing function depends on the distance between two individuals of the population. The simplified version is the following form (Goldberg & Richardson, 1987):

$$sh(dist(x,y))=\begin{cases} 1-\left(\frac{dist(x,y)}{\delta_{sh}}\right)^{\alpha} \; if \; dist(x,y)<\delta_{sh} \\ O \; otherwise \end{cases}$$

α : constant

δ_{sh} : dissimilarity threshold

The distance function can be defined in the genotypic or phenotypic space search (Fonseca & Fleming, 1995) or their combination (Horn, 1997).

Mahfoud (1995) applied the principle of perfect discrimination of the niches giving two main consequences:

- each individual in a given niche, regardless of the distance measure employed, is always closer to every individual of its own niche than to any individual of another niche,
- the difference measure is able to determine whether two individuals are members of the same niche.

The author concludes that the sharing technique is most effective in cases of no overlap niches.

3.2.2. Clearing technique

The clearing technique(Petrowski, 1997) is a niching method based on the sharing ecological theory. It is applied after evaluating the fitness of individuals and before applying the selection operator. As in the sharing method, the clearing algorithm uses a dissimilarity measure between individuals to determine if they belong to the same subpopulation or not. In contrast, the clearing procedure gives all the resources of a niche to a single individual : the winner. The winner takes all rather than sharing resources with the other individuals of the same niche.

The clearing procedure is less complex and more compatible with elitist strategies than the sharing technique (Petrowski, 1997).

4. Our approach : multiple query niches evaluation

4.1. The genetic relevance optimisation process

The goal of our GA is to find an optimal set of documents which best matched the user's needs. The GA attempts to involve, generation by generation, a population of query niches towords those improving the outcome of the system. The retrieval process as shown in figure 1, is based on an iterative feedback evaluation of query niches. A niche represents a set of individual queries exploring a specific region of the document space according to their evaluation results. The genotype representation of an individual query is of the form Q_u (q_{u1}, q_{u2}, …, q_{uT}). Each gene corresponds to an indexing term or concept. Its value or *locus* is represented by a real value and defines the importance of the term in the considered query. Initially, a term weight can be computed by any query term weight scheme ; it will then evolve through the generations. In our case, we used the following formula :

$$q_{ui} = \frac{(1+\log(tf_{ui}))*\log(\frac{N}{n_i})}{\sqrt{\Sigma_{k=1}^{T}((1+\log(tf_{uk}))*\log(\frac{N}{n_k})))^2}}$$

N: total number of documents

n_i : number of documents containing tem t_i

tf $_{uk}$: frequency of term t_k in documenu u

Initially all the individual queries are grouped in a single niche.The phenotype of an individual query is translated by its evaluation results in the IRS.

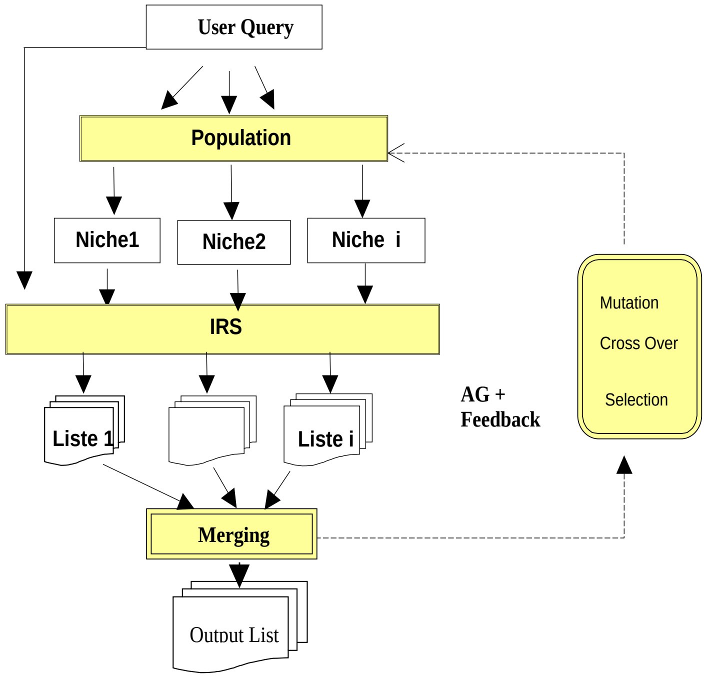

Figure 1 : **The genetic retrieval process**

The retrieval process runs using the following algorithm:

```
Begin
  Submit the initial query and do the search
  Judge the top thousand documents
  Build the initial population
  Repeat
      For each niche of the population
          do the search
          build the local list of documents
      Endfor
  Build a merged list
  Renew the niches
  Judge the top fifteen documents
  Compute the fitness of each individual query
  For each niche $N^{(s)}$ of the population
  Repeat
      parent1= Selection ($N^{(s)}$)
      parent2= Selection ($N^{(s)}$)
      Crossover (Pc , parent1, parent2,son)
      Mutation (Pm , son, sonmut)
      Add_Niche (sonmut,$N^{(s+1)}$)
  Until Niche_size ($N^{(s+1)}$) = Niche_size ($N^{(s)}$)
Until a fixed number of feedback iterations
End
```

4.2. The niching method

In the current study, we applied the sharing technique to build the niches. Our choice is influenced by the fact that we attempt to widely explore the document space. We hope that the analysis of our first experiments using this technique will result in suitable utilities for further exploitation of other niching techniques such as the clearing one.

Regardless of the niching method used, the fitness function must be correlated with the standard goodness measure in IR that is average and precision. Considering this characteristic, we propose two distinct fitness function formulations. Each is related to a specific strategy of formation of the niches.

4.2.1. Niching using genotypic sharing

In this case, a niche is a set of individual queries having closed genotypes. According to the general form of Goldberg & Richardson sharing function form, our sharing function is the following:

$$sh(dist(Q_u^{(s)},Q_v^{(s)}))=\begin{cases}1\, if\;\; dist(Q_u^{(s)},Q_v^{(s)})<\delta \\ 0\, otherwise\end{cases}$$

$Q_u^{(s)}$: individual queyr at the generation s of the GA

dist: Euclidian distance

δ: niching threshold ($\delta > 0$)

The function has the following properties:

1. $0\leq sh(dist(Q_u^{(s)},Q_v^{(s)})\leq 1$

2. $sh(0)=1$

3. $\lim_{dist(Q_u^{(s)},Q_v^{(s)})\rightarrow\infty} sh(dist(Q_u^{(s)},Q_v^{(s)}))=0$

Furthermore, the niches are perfectly distinct.

The fitness function is then computed using the formula :

$$Fitness(Q_u^{(s)})=\frac{QFitness(Q_u^{(s)})}{\sum_{Q_v^{(s)}\in Pop} sh(dist(Q_u^{(s)},Q_v^{(s)}))}$$

where:

$$QFitness(Q_u^{(s)})=\frac{\frac{1}{|Dr|}*\sum_{dr\in Dr} J(dr,Q_u^{(s)})}{\frac{1}{|Dnr|}*\sum_{dnrDnr} J(dnr,Q_u^{(s)})}$$

dr: relevant document

dnr: irrelevant document

Dr: set of relevant documents retrieved across the GA generations

Dnr: set of irrelevant documents retrieved across the GA generations

$J(D_j , Q_u^{(s)})$: Jaccard measure

Thus, the more query retrieves relevant documents, the fitter the query is. This fitness function would favour the reproduction of queries that are close to relevant documents and away from non relevant documents.

4.2.2. Niching using phenotypic sharing

In this case, formation of the niches is based on the evaluation results of their individual query members rather on their genotypic similarity. The niche structure is defined according to the coniche operators as follows:

$$(Q_u^{(s)}\equiv_N Q_v^{(s)})\Leftrightarrow(\left|(Ds(Q_u^{(s)},L))\cap(Ds(Q_v^{(s)},L)\right|)>Coniche\_Limit)$$

$Q_u^{(s)}$: individual query at generation (s) of the GA

$Ds(Q_u^{(s)},L)$: the L top documents retrieved by $Q_u^{(s)}$

Coniche _ Limit: the min number of common documents retrieved by queries of the same niche

Thus, queries belonging to the same niche have close evaluation results.

In order to maintain distinct niches, we assume the setting of an individual query once to the lowest capacity niche. The fitness function is computed using a formula built on the Guttman model (Guttman, 1978):

$$Fitness(Q_u^{(s)})=1+\frac{\sum_{dr\in Dr(s),\, dnr\in Dnr(s)} J(Q_u^{(s)},dr)-J(Q_u^{(s)},dnr)}{\left|\sum_{dr\in Dr(s),\, dnr\in Dnr(s)} J(Q_u^{(s)},dr)-J(Q_u^{(s)},dnr)\right|}$$

J: Jaccard measure

$Dr^{(s)}$: set of relevant documents retrieved at the generation(s) of the GA

$Dnr^{(s)}$: set of non relevant documents retrieved at the generation(s) of the GA

dr: relevant document

dnr: irrelevant document

4.3. Genetic operators

The genetic operators defined in our approach are not classical ones as they are not based on the basic structure proposed in GA theory (Goldberg, 1989). They have been adapted to take advantage of techniques developed in IR. Thus, we qualify them as knowledge based operators. In addition, they are restrictively applied to the niches in order to focus the search in the corresponding directions of the document space.

- *Selection*

The selection procedure is based on a variant of the usual roulette wheel selection (Goldberg, 1994). It consists essentially of assigning to every individual of the population a number of copies in the next generation, proportional to its relative fitness. Comparatively, to the roulette wheel selection, the number of clones generated by each individual is closer to the corresponding relative fitness value.

- *Crossover*

The crossover procedure is applied to a pair of individuals that are selected in the same niche, according to the crossover probability Pc . We define a crossover based on term weight, with no crossing point. It enables the modification of the term weights according to their distribution in the relevant and in the non-relevant documents. Let us consider $Q_u^{(s)}$ and $Q_v^{(s)}$ two individuals selected for crossover. The result is the new individual $Q_p^{(s)}$ defined as:

$$\begin{tikzcd}[column sep=small]
Q_u^{(s)}(q_{u1}^{(s)}, q_{u2}^{(s)}, \ldots, q_{uT}^{(s)}) \arrow[dr] & & Q_v^{(s)}(q_{v1}^{(s)}, q_{v2}^{(s)}, \ldots, q_{vT}^{(s)}) \arrow[dl] \\
& Q_p^{(s+1)}(q_{p1}^{(s+1)}, q_{p2}^{(s+1)}, \ldots, q_{pT}^{(s+1)}) &
\end{tikzcd}$$

$$q_{pi}^{(s+1)} = Max(q_{ui}^{(s)}, q_{vi}^{(s)})\ if\ weight(t_i, Dr^{(s)}) \geq weight(t_i, Dnr^{(s)})$$
$$Min(q_{ui}^{(s)}, q_{vi}^{(s)})\ otherwise$$

We defined:

$$weight(t_i, D) = \sum_{dj \in D} d_{ji}\ ,$$

d_{ji} : term weight of t_i in d_j
D : a set of documents

In other words, if the weight of term t_i in the set of relevant documents is higher than its weight in the set of non-relevant documents, this term is retained as significant and the highest weight among ($q_{ui}^{(s)}$, $q_{vi}^{(s)}$) is assigned to this term in the new query $Q_p^{(s+1)}$. Otherwise, it is assigned the lowest weight the new query.

- *Mutation*

This consists essentially of exploring the terms occurring in the relevant documents in order to expand and/or reweight the query selected for the mutation. Let us consider $Q_u^{(s)}$ as the selected individual query and $Lmut^{(s)}$ as the set of terms from $Dr^{(s)}$ the relevant documents retrieved at the last generation of the GA. The mutation will alter genes of the selected individual on the basis of the $Lmut^{(s)}$ terms and on the probability *Pm* . The $Lmut^{(s)}$ terms are sorted according to a score value calculated as follows:

$$Score(t_i) = \frac{\sum_{dj \in Dr^{(s)}} d_{ji}}{\left\| Dr^{(s)} \right\|}$$

The mutation operation is done as follows:

```
1. For each term $t_i$ in $Lmut^{(s)}$
2. If (random(p)<Pm) then
3. $q_{ui}^{(s)} = average(Q_i^{(s)})$
4. Endif
5. Endfor
```

random(p) generates a random number p in the range [0..1]. The average function is computed as follows:

average ($Q_u^{(s)}$) = $\frac{\sum_j^T q_{ui}^{(s)}}{nq_{ui}^{(s)}}$

where $nq_{ui}^{(s)}$ is the number of $q_{ui}^{(s)} \neq 0$ in $Q_u^{(s)}$.

4.4. Merging method

At each generation of the GA, the system presents to the user a limited list of new documents. These documents are selected from the whole ones retrieved by all the individual queries of the population, using a specific merging method. We investigate two main methods for building the merged list depending on the different sharing techniques.

- *genotypic sharing:* the merged list is built using the following formula

$$\mathrm{Re}l(D_j) = \sum_{Q_u^{(s)} \in Pop^{(s)}} RSV(Q_u^{(s)}, D_j)$$

$RSV(Q_u^{(s)}, D_j)$: *RSV (Retrieval Status Value) of the document at the generation (s) of the GA*

$Pop^{(s)}$: *population at the generation (s) of the GA*

Thus the assumed relevance value of a document depends on the corresponding relevance status values resulting from the individual query evaluation.

- *phenotypic sharing:* the rank order of the documents is computed as follows

$$\mathrm{Re}l(D_j) = \sum_{N_j^{(s)} \in Pop^{(s)}} \sum_{Q_u^{(s)} \in N_j^{(s)}} Fitness(Q_u^{(s)**} * RSV(Q_u^{(s)}, D_j)\ )$$

$Q_u^{(S)**}$: *individual queries characterised by a fitness value higher than the average fitness of $Pop^{(s)}$*

$N_j^{(s)}$: *jth niche at the current generation s of the GA*

In the case of this merging method, the merging process doesn't deal with the evaluation results of the whole individual queries but only the fittest ones. Thus, we attempt to emphasise the ranking value of documents resulting from the evaluation of *good* queries.

5. Experimental results

The experiments were carried out on a sub-collection of TREC corpus : AP88 using 144186 documents, 25 queries and sets of relevant documents of each query . The experiments were run using the Mercure IRS that processes the spreading activation. The main goal of these experiments was to evaluate the effectiveness of our GA for a multiple query evaluation in comparison with a traditional single query evaluation. We also compare the effect of both phenotypic and genotypic sharing and finally measure the effectiveness of the genetic operators proposed. The conventional measures ie recall and precision are used in the evaluation. Because of the multiple iteration aspect of the search and the use of relevance judgements, the results reported in the paper are based on a residual ranking evaluation (Chang & al, 1971). This method is used to evaluate the effectiveness of relevance feedback methods. In this method all the documents previously judged are removed from the document rankings produced by both the initial query, which corresponds to the iteration 0 in our algorithm, and the feedback query which corresponds to iteration 1 in our algorithm. Precision and recall are computed for these and then for both residual lists of documents. In the case of multiple iteration the comparison is done in the same way between the residual documents retrieved at iteration *(i)* to the residual documents retrieved at iteration *(i+1).* This tells us how much we gaines by doing the next iteration of the GA. In order to measure the effect of the GA, we report at each iteration the number of relevant documents in the top 15 retrieved and, written in brackets, the cumulative number of relevant documents retrieved at this point.

Prior experiments (Boughanem & al, 1999) (Tamine & Boughanem, 2001) allowed us to evaluate the main parameters: population size varying from 4 to 6, crossover probability = 0.7, mutation probability = 0.07, coniche limit = 9, similarity threshold= 0.3.

5.1. Genetic multiple query evaluation Vs single query evaluation

At this level, we address the question of how well our genetic combination performs relative to a single query evaluation. For this aim, we compare the performance results obtained from two distinct runs:

- the first one based on a genetic combination of multiple query evaluation results as described above
- the second one based on a classic single query evaluation as performed in Mercure IRS

In order to make sense of our comparative evaluation, we consider that an iterative single query evaluation process may be based on scanning the overall output list, from top in direction to bottom, using sub-lists presented to the user. This means that we analyze at each iteration the following sub-list of documents (15 documents in the case of our experiments) ordered after the above list presented to the user according to the output list.

Finally, we compare the retrieval performance of residual lists issued from the same iteration of both single query evaluation and genetic combination processes.

The evaluation results obtained using genotypic sharing and phenotypic sharing are shown respectively in table 1.a and table 1.b.

Genotypic sharing					
	Iter1	**Iter2**	**Iter3**	**Iter4**	**Iter5**
Single query evaluation	110(110)	92(203)	82(285)	65(351)	64(412)
Genetic multiple query evaluation	177(177)	114(160)	93(215)	69(266)	56(296)
Improvement	6%	8%	15%	15%	10%

Table 1.a: retrieval performances using genotypic sharing

Phenotypic sharing					
	Iter1	**Iter2**	**Iter3**	**Iter4**	**Iter5**
Single query evaluation	110(110)	92(203)	82(285)	65(351)	64(412)
Genetic multiple query evaluation	180(180)	88(266)	97(366)	75(442)	78(520)
Improvement	63%	32%	28%	25%	26%

Table 1.b: retrieval performances using phenotypic sharing

These experimental results indicate that our approach yields large improvements over a traditional simple evaluation process. More precisely, we notice that the improvements vary from 6% to 15% in the case of applying a genotypic sharing and vary from 5% to 15% in the case of applying a phenotypic sharing.

This experiment globally validates our approach. The experiments presented below compare the effect of the niching techniques and also measure the effect of knowledge based operators on the retrieval results.

5.2. Comparative evaluation of the sharing techniques

This experiment compares the sharing techniques proposed using the AP88 collection test. We report in table 2 the number of relevant documents in the top 15 retrieved at each iteration of the GA and the cumulative number of relevant documents retrieved at that point, using both genotypic sharing and phenotypic sharing.

	Iter1	**Iter2**	**Iter3**	**Iter4**	**Iter5**
Genotypic sharing	177(177)	114(291)	93(384)	69(453)	56(510)
Improvement	38%	41%	24%	25%	22%
Phenotypic sharing	180(180)	88(268)	97(366)	75(442)	78(520)
Improvement	63%	32%	28%	25%	26%

Table 2 : Comparative evaluation of the sharing techniques

Table 2 reveals that the phenotypic sharing technique is more effective than the genotypic one. More precisely, the cumulative number of relevant documents retrieved at the fifth generation of the GA is 510 using the genotypic sharing and 520 using the phenotypic sharing. The number of relevant documents retrieved by iteration is also generally higher when using phenotypic sharing.

These results are in accordance with previous analyses presented in (Mahfoud, 1995) (Talbi, 1999) on the effectiveness of the phenotypic sharing technique. The main reason might be due to *the meaning distance* between the genotypic individual representation and its significant phenotypic one.

5.3. Effect of the niching technique

The main goal of using niching technique is to reach different optima for a specific optimisation problem. In the context of our study, niching would enable the recall of relevant documents with quite different descriptors. In order to evaluate its precise effect on the search results, we have organised the query collection test into bins. Each bin is characterised by a corresponding average similarity value between relevant documents in fixed intervals [20 25[, [25 30[, [30 35[. We then plotted the histogram presented in figure 2. The x-axis represents the document bins, the y-axis represents the cumulative number of relevant documents retrieved at the fifth generation of the GA.

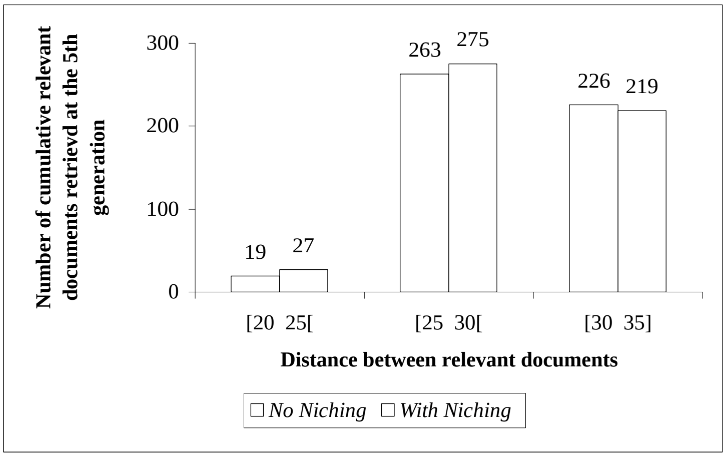

Figure 2: Effect of the niching technique

It can be seen that the niching technique improves the results for the first and the second bin with respectively 42% and 45% above the baseline. In contrast, the performance decreases in the case of the third bin. A reason for this could be that due to the relatively large distance between relevant documents, the convergence of the GA becomes slow.

Considering this assumption, we have developed this experimentation by running the 6th iteration of the GA just on the third bin of queries. Table 3 shows the effect of the niching technique on the cumulative number in the top 15 retrieved at this iteration.

Query number	No niching	With niching
22	64	*83*
11	41	*40*
25	14	*14*
10	40	*40*
16	6	*9*
12	37	*33*
21	9	*10*
17	55	*53*
14	16	*14*
	282	*296*

Table 3 : Effect of niching at the 6th iteration of the GA

We notice clearly that the results are better when using the niching technique at the following iteration of the GA (4,9 % of improvement). This suggests that in order to increase the convergence of the GA, it might be interesting to use more suitable combination between the coniche operator definition and prior user relevance judgements.

5.4. Effect of the knowledge based operators

Table 4 compares the results of the GA using the knowledge based operators and the classical ones. The classical crossover is based on the classical GA crossover operator. Let us consider $Q_u^{(s)}$ and $Q_v^{(s)}$ two individuals selected for crossover with *c* as the crossing point and $Q_u^{(s+1)}$ and $Q_v^{(s+1)}$ as the new individuals resulting from the classical crossover. This operator is defined as follows:

$$Q_{p1}^{(s+1)}\,(\,q_{u1}^{(s+1)},\, q_{u2}^{(s+1)},\quad q_{vc}^{(s),} q_{vc+1}^{(s)}\; \ldots.;\; q_{vT}^{(s)})$$

$$Q_{p2}^{(s+1)}\,(\,q_{v1}^{(s+1)},\, q_{v2}^{(s+1)},\, q_{uc}^{(s),} q_{uc+1}^{(s)},\ldots,\, q_{uT}^{(s)})$$

$$if(c<i)then\begin{cases} q_{p1i}^{(s)}=q_{ui}^{(s)} \\ q_{p2i}^{(s)}=q_{vi}^{(s)} \end{cases} \quad else\begin{cases} q_{p1i}^{(s)}=q_{vi}^{(s)} \\ q_{p2i}^{(s)}=q_{ui}^{(s)} \end{cases}$$

In classical mutation, the genes are mutated by modifying their weight arbitrarly.

	Iter1	**Iter2**	**Iter3**	**Iter4**	**Iter5**
Class Op	171 (171)	79 (250)	65 (315)	65 (380)	68(449)
Knowl. Op	180(180)	88 (268)	97 (366)	75 (442)	78 (520)
Improvement/ Cum_Doc	**5,2%**	**7,2%**	**16%**	**16%**	**15%**

Table 4: Results for knowledge based operators vs. classical operators

We clearly notice that the knowledge-based operators are more effective than the classical ones. Indeed, both the number of relevant documents and the cumulative number of documents are much higher when applying enchanced operators than classical ones with an improvement of 15% at the fifth iteration. This supports our intuition behind the interesting use of information retrieval techniques when performing genetic transformations on the individual queries.

Conclusion

This paper proposes a novel approach to resolve the multimodal problem of relevance optimisation in IR. The approach is mainly based on the integration of a niching technique in a GA which performs a multiple query evaluation. The GA is also characterised by using operators adapted to the context of the retrieval task.

Several different kinds of experiments carried out on TREC sub-collections validate our approach. Indeed, the results presented demonstrate the effectiveness of our genetic approach in performing multiple query evaluation. Furthermore, we compared the results produced by different sharing techniques and showed the global effectiveness of the niching method. In this study, we also demonstrated the interesting use of knowledge domain to develop the structure of the genetic operators.

In the future, we plan to provide a formal definition of the niche concept and propose more suitable merging formula. In addition, we plan to test and compare other niching techniques in some TREC tasks.

References

J E.Baker (1985). Adaptive Selection Methods for Genetic Algorithm, in Proceedings of the first International Conference on Genetic Algorithm (ICGA) pp 101-111.

D. Beasly, D.R Bull & R. R Martin (1993). A sequential niche technique for multimodal function optimization, Evolutionary Computation, 1(2) : pp 101-125.

N. J. Belkin, C. Cool, W. Bruce Croft, J. P. Callan (1993). Effect of multiple query representations on information retrieval system performance. In Proceedings of ACM SIGIR, Conference on Research and Development in Information Retrieval , pp 339-346, Pittsburgh.

M. Boughanem (1997). Query modification based on relevance backpropagation, In Proceedings of the 5th International Conference on Computer Assisted Information Searching on Internet (RIAO'97), pp 469-487, Montreal.

M. Boughanem, C. Chrisment & L.Tamine (1999) :Genetic Approach to Query Space Exploration. Information Retrieval Journal, Vol 1 N°3 , pp175-192.

YK Chang , GC. Cirillo and J. Razon (1971). Evaluation of feedback retrieval using modified freezing, residual collection and test and control groups. In : The Smart Retrieval System: Experiments in Automatic document processing, Prentice-Hall Inc, chap 17, pp 355-370.

K. A Dejong (1975). An analysis of the behavior of a class of genetic adaptive systems, Doctocal dissertation University of Michigan,. Dissertation abstracts International 36 (10), 5140B. University Microfilms N°76-9381.

S. Dumais (1994). Latent Semantic Indexing (LSI), TREC3 report. In Proceedings of the 3rd Conference on Text Retrieval Conference (TREC) pp 219-230.

C.M Fonseca & P. J Fleming (1995). Multi-objective genetic algorithms made easy: selection, sharing and mating restrictions, In IEEE International Conference in Engineering Systems: Innovations and Application, pp 45-52, Sheffield, UK.

D.E. Goldberg & Richardson (1987). Genetic algorithms with sharing for multimodal function optimization, in Proceedings of the second International Conference on Genetic Algorithm (ICGA) , pp 41-49.

D.E. Goldberg (1989) : Genetic Algorithms in Search, Optimisation and Machine Learning, Edition Addison Wesley 1989.

M. Gordon (1988) . Probabilistic and genetic algorithms for document retrieval, Communications of the ACM pp 1208-1218.

L. Gutman (1978). What is Who What in Statistics. The Statistician, 26 pp 81 :107, 1978

D. Harman (1992). Relevance feedback revisited : In Proceedings of ACM SIGIR, Conference on Research and Development in Information Retrieval, pp 1-10.

D. Haines & W.B Croft (1993). Relevance Feedback and Inference Networks, Conference on Research and Development in Information Retrieval (SIGIR), pp 2-11, 1993.

J. Hollan (1962). Concerning Efficicent Adaptive Systems.In M.C Yovits, G.T Jacobi, &G.D Goldstein(Eds) Self Organizing Systems pp 215-230 Washinton : Spartan Books, 1962.

J. Horn (1997). The nature of niching : Genetic algorithms and the evolution of optimal cooperative populations, PhD thesis, university of Illinois at Urbana, Champaign.

J.T Horng & C.C Yeh (2000). Applying genetic algorithms to query optimisation in document retrieval, In Information Processing and Management 36(2000) pp 737-759.

Katzer , M.J. McGill, J.A. Tessier, W. Frakes and P. DasGupta (1982). A study of the overlap among document representations. Information Technology : Research and Development, 1 (4) : pp 261-274.

D.H Kraft, FE Petry, B.P Buckles and T. Sadisavan (1995). Applying genetic algorithms to information retrieval system via relevance feedback, In Bosc and Kacprzyk J Eds, Fuzziness in Database Management Systems Studies in Fuzziness Series, Physica Verlag, pp 330-344, Heidelberg, Germany.

K. L Kwok (1995). A network approach to probabilistic information retrieval, ACM transactions on information systems, vol 13 N°3, pp 324-353.

J. H. Lee (1997). Analyse of multiple evidence combination , In Proceedings of ACM SIGIR, Conference on Research and Development in Information Retrieval pp 267-275.

S.W. Mahfoud (1995). Niching methods for genetic algorithms, PhD thesis, university of Illinois at Urbana, Champaign, 1995.

MCGill, Koll & Norreeault (1979). An evaluation of factors affecting document ranking by IR systems, Syracuse, Syracuse university school of information studies.

A. Petrowski (1997) . A clearing procedure as a niching method for genetic algorithms. In the Proceedings of the IEE International Conference on Evolutionary Computation (ICEC), Nagoya, Japan.

S.E Robertson & K. Sparch Jones (1976). *Relevance Weighting for Search Terms,* Journal of The American Society for Information Science (JASIS), Vol 27, N°3, pp 129-146.

S. E. Robertson (1977). The probability ranking principle in IR, Journal of documentation 33 (4), pp 294 – 304.

S. E Robertson & S. Walker (1997). On relevance weights with little relevance information, In Proceedings of the 20th annual international ACM SIGIR conference on research and development, pp 16-24, 1997.

Rocchio(1971). Relevance Feedback in Information Retrieval, in The Smart System Experiments in Automatic Document Processing, G.Salton, Editor, Prentice-Hall, Inc., Englewood Cliffs, NJ, pp 313-23, 1971.

G. Salton (1968). Automatic Information and Retrieval, Mcgrawhill Book Company, N. Y., 1968.

E.G Talbi (1999). Métaheuristiques pour l’optimisation combinatoire multi-objectifs : Etat de l’art, Rapport CNET (France Telecom) Octobre 1999.

L.Tamine (2000). Optimisation de requêtes dans système de recherche d'information, approche basée sur l'exploitation de techniques avancées de l'algorithmique génétique. Doctorat thesis, University Paul Sabatier, Toulouse, France

L. Tamine & M. Boughanem (20001). Un algorithme génétique spécifique à une évaluation multi-requêtes dans un système de recherche d'information, Journal Information Intelligence et Interaction, Vol 1 N°1, september 2001.

H. Turtle & W.B. Croft (1991). Evaluation of an inference network-based retrieval model, ACM Transactions on information systems, 9, 3: pp 187-222.

J.J Yang & R.R Korfhage (1993). Query optimisation in information retrieval using genetic Algorithms, in Proceedings of the fifth International Conference on Genetic Algorithms (ICGA), pp 603-611, Urbana, IL.