Journal of
the Brazilian Computer Society
a SpringerOpen Journal

RESEARCH Open Access

Findings on ranking evaluation functions for feature weighting in image retrieval

Sergio F da Silva[1*], Letricia PS Avalhais[1], Marcos A Batista[2], Celia AZ Barcelos[3] and Agma JM Traina[1]

Abstract

Background: There are substantial benefits to be gained from ranking optimization in several information retrieval and recommendation systems. However, the analysis of ranking evaluation functions (REFs), which play a major role in many ranking optimization models, needs to be further investigated. An analysis of previous studies that investigated REFs was performed, and evidence was found which indicated that the choice of a proper REF is context sensitive.

Methods: In this study, we analyze a broad set of REFs for feature weighting aimed at increasing the image retrieval effectiveness. The REFs analyzed sums ten and includes the most successful and representative REFs from the literature. The REFs were embedded into a genetic algorithm (GA)-based relevance feedback (RF) model, called WLSP-C±, aimed at improving image retrieval results through the use of learning weights for image descriptors and image regions.

Results: Analyses of precision-recall curves in five real-world image data sets showed that one non-parameterized REF named F5, not analyzed in previous studies, overcame recommended ones, which require parameter adjustment. We also provided a computational analysis of the GA-based RF model investigated, and it was shown that it is linear in regard to the image data set cardinality.

Conclusions: We conclude that REF F5 should be investigated in other contexts and problem scenarios centered on ranking optimization, as ranking optimization techniques rely heavily on the ranking quality measure.

Keywords: Rank learning; Ranking evaluation functions; Content-based image retrieval; Genetic algorithms

Background

Ranking optimization research studies have fostered widespread developments in information retrieval and recommendation systems [1-6]. Ranking optimization techniques can be grouped into three main classes: rank learning [2,4,5], rank aggregation (also known as data fusion) [7-10] and ranking (or list) diversification [1,11,12]. Rank learning relies on supervised queries, relevance feedback or context information to achieve an adequate model to rank items like web pages, images, etc. Normally, rank aggregation is an unsupervised method that relies on multi-criteria ranks and tries to combine them to produce a consensus rank. On the other hand, ranking or list diversification aims at balancing 'precision' and 'diversity' to reflect a broad spectrum of user interests concerning items.

Rank learning tasks are generally stated as optimization problems: to find the best model (or the best adjustment in a given model) according to some representation to rank items. Given its general formulation, solutions of rank learning normally apply a search method guided by some ranking evaluation function. Ranking evaluation functions (REFs) are normally computed with a basis on supervised queries or user relevance feedback (RF). These REFs evaluate models or adjustments according to the effectiveness of the ranking produced. In regard to search methods, most research studies have employed evolutionary algorithms (EAs). The EA flexibility enables the modeling of rank learning in many ways, such as through ranking function discovery [5,13,14], weight and parameter learning [15-19], among others. Independent to the model representation, a proper evaluation function is very important for the effectiveness and efficiency of EAs.

Although REFs have been shown to have applied a major rule to rank learning almost a decade ago [13,15-17], in recent studies, little attention has been given to the design

*Correspondence: sergio.f.silva@gmail.com
[1] Federal University of Goiás, Catalão, 75604-020 Goias, Brazil
Full list of author information is available at the end of the article

and selection of more appropriate ones. Researchers have chosen popular REFs and applied them to new contexts and models without any theoretical or empirical evidence about its suitableness. Moreover, few studies have focused on rank learning for image retrieval tasks, and the existing ones are not deep enough and do not cover all the spectrum of models employed in this sector.

López-Pujalte et al. [15-17] have studied the problem of adapting document descriptions through learning terms, weights and parameters in matching functions applied to information retrieval. These researches investigated mainly the issue of different REFs as fitness functions for genetic algorithms (GAs) in relevance feedback. By analyzing the mean precision in three levels of recall, these studies showed that the results effectiveness varied widely depending on which REF is used. Also, in these studies, it was found that utility theory-based ranking evaluation functions (UTB-REFs) comprises the most adequate kind of REF for rank learning applications. Moreover, the REF named F4 in this present study was recommended by López-Pujalte et al. in [17] as a promising one.

Fan et al. [13] compared seven UTB-REFs on ranking function discovery for Web search using genetic programming (GP). Their experiments on a large Web Corpus revealed that some UTB-REFs, named F9, F7, F8 and F3 in this present study, were more effective in guiding the GP search than others which were analyzed. In a following investigation, Fan et al. [20] used the UTB-REF named F10 in this paper, with the aim of increasing the precision of information retrieval in two steps: first, by discovering new ranking functions using genetic programming; second, by combining document retrieval scores of different ranking functions using genetic algorithms. The use of UTB-REF F10 was justified since it is a standard performance measure used in information retrieval studies.

Torres et al. [5] used GP to discover functions to combine different descriptors for content-based image retrieval (CBIR) tasks. Their method relies on a training set containing query images together with the relevant images to each query image and, obviously, a REF that guides the GP search towards a proper combination function. In this context, the authors tested seven UTB-REFs as fitness functions in the GP - the same UTB-REFs used by Fan et al. in [13]. The UTB-REFs that produced the best results are named F6, F7 and F4 in this paper. Ferreira et al. [14] proposed a similar method of [5] using RF instead of a training set of queries. This study does not compare REFs and uses the UTB-REF F4, due to its promising results in [5].

Stejić et al. [19] used a GA-based RF model to improve image retrieval results by applying learning weights to image descriptors and image regions (WLSP-C± model). This study presented promising approaches such as the concept of local similarity patterns (LSP) and the use of continuous positive and negative weights modeling relevance and undesirability of visual features. In spite of the promising features of the model, the authors did not provide an effective mechanism for learning a proper set of weights. The use of the *R*-precision measure without any other REF analysis is the most critical aspect of the Stejić et al. research, as other studies had shown that UTB-REFs are more appropriate for such ranking modeling.

Silva et al. [18] extended the WLSP-C± model by Stejić et al. [19] proposing a new UTB-REF in substitution to the *R*-precision measure used as the objective (fitness) function into GA. Their results showed a significant improvement in the image retrieval precision and in efficiency as the proposed UTB-REF speed up the GA search in direction of optimal solutions.

As we can observe from the studies reported, there is no consensus about which is the best REF for many of the applications, and many studies have overlooked the REF analysis. Even for the same task, there is no consensus about the best REF, as we can see from the REF analysis performed in the studies by Fan et al. [13] and Torres et al. [5] that employed the same set of REFs. In this way, we will show that there is space for development in this issue and that new studies should consider the analysis of broad sets of REFs, due to the fact that a proper choice should be context-sensitive.

In this paper, we used the WLSP-C± model proposed by Stejić et al. [19] and used in [18] to investigate a broad set of REFs for feature weighting aimed at improving image retrieval performance. The choice of WLSP-C± model was motivated by its promising results. The REFs were applied as fitness functions into a specialized GA for learning weights. Analyses of precision-recall curves in five real-world image data sets showed that the REF design applies a key role regarding the effectiveness and efficiency of the WLSP-C± model. Also, we found that the non-parameterized REF proposed in [18] and named F5 in this present paper overcame recommended ones, which require parameter adjustment. This result indicates that the REF F5 should be investigated in other contexts and problem scenarios centered on ranking optimization mainly for image retrieval, as ranking optimization techniques rely heavily on the ranking quality measure.

The remainder of this paper is organized as follows. The ‘Methods’ Section describes the methodology employed for the analysis of REFs on the WLSP-C± model. The ‘Results and discussion’ Section compares a broad set of REFs for feature weighting aimed at improving image retrieval and provides a computational complexity analysis of the model. The ‘Conclusions’ Section concludes the paper highlighting the main findings and implications of the present research.

Methods

In this study, we used the WLSP-C± model [19] to investigate a broad set of ranking evaluation functions. The weights of WLSP-C± model were optimized using a GA-based RF mechanism reported in [18]. This methodology is illustrated in Figure 1. We stored into a database all the images considered for a given image searching task. The image database is linked to the module of feature extraction. The output data of the feature extraction module is a structure containing the identification code and the feature vectors of color, shape and texture for each image of the database. These data (identification code/features) are stored in the feature database.

When the user carries out a search, feature vectors of color, shape and texture are extracted from the query image by the feature extraction module and compared, through similarity measures, found in the image feature vectors from the range of images stored in the database. The similarity measure module returns a similarity value $S_I(q, i)$ for each image in the database, in relation to the query image. Then, the images are sorted in decreasing order of similarity (ranking) and the first samples are shown to the user. Not satisfied with the result of the search, the user can provide feedback, indicating to the system the relevant images according to his/her point of view. Based on the user's feedback, the GA-based relevance feedback mechanism adjusts the similarity measure according to the user's criteria through image feature vector weighting (ω_F) and region weighting (ω_R). n_g corresponds to the number of generations for the genetic algorithm.

The retrieval process is based on the *local similarity pattern*, where the image areas are uniformly partitioned into regions, and the similarity between images is measured by corresponding region similarities. Similarity between regions, and therefore between images, is computed through three feature vectors (F) encoding properties of color, shape and texture, represented by color moments, edge direction histogram and texture neighborhood, respectively. The distance between pairs of color feature vectors is computed by *Euclidean distance*, while distances between pairs of shape and texture feature vectors are computed by *city-block distance*.

To make comprehension easier, we present in the next subsections a detailed description of the WLSP-C± model and the GA-based RF mechanism. Then, we describe the analyzed ranking evaluation functions and also the employed image data sets.

WLSP-C± model

Let q be the query image, I be the image data set, i be an image belonging to I, r be an image region belonging to R such that $R = \{r_1, r_2, \ldots, r_m\}$ is given by a rectangular tiled partition of i, and f be an image feature vector. The image similarity measure is given by Equation 1, where $S_F(q, i, r, f)$ represents the similarity between the images q and i, in relation to the feature vector f in the region r; $\omega_F(r, f)$ weighs with real values in range $[-1, 1]$ the importance of f in the region r and is responsible for the S_F normalization; $\omega_R(r)$ weighs with real values in range $[-1, 1]$ the importance of the image region r; and finally, $S_I(q, i)$ gives the overall image similarity between q and i.

$$S_I(q,i) = \sum_{r \in R} \left(\omega_R(r) \sum_{f \in F} \left(\omega_F(r,f) S_F(q,i,r,f) \right) \right). \quad (1)$$

The WLSP-C± model is optimized by fitting the weights $\omega_R(r)$ and $\omega_F(r, f)$, so that the retrieval accuracy according to the query image and the set of relevant

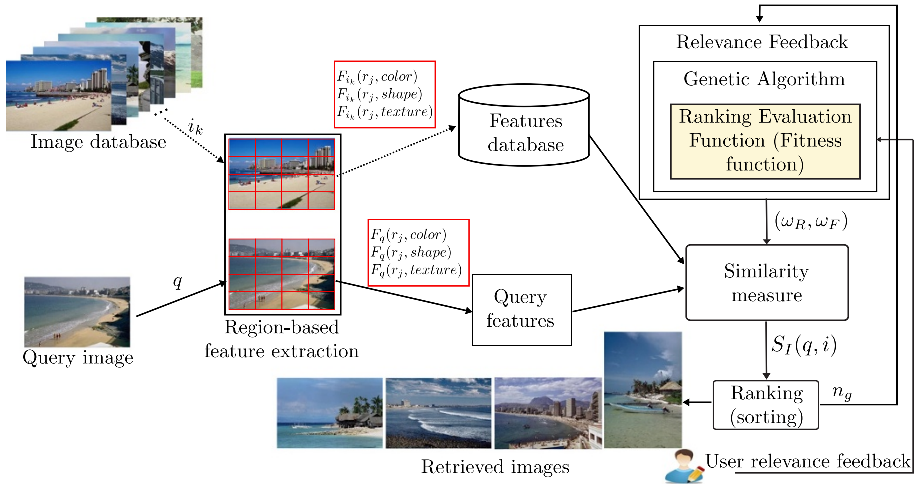

Figure 1 Outline of the employed methodology.

images chosen by the user is maximized. As in [19] and [18], we solve this optimization problem using a real-code GA that infers weights in the range $[-1, 1]$. Continuous negative and positive allows for the mapping of the user's concepts of relevance, irrelevance and undesirability of image visual properties producing superior results than positive weights alone as shown in [19]. Since we found the best results with the WLSP-C± model, we did not analyze in this study the other models proposed by Stejić et al. in [19].

The GA-based RF mechanism

Our RF mechanism relies on a GA designed and adjusted for learning weights in the paper [18]. Algorithm 1 describes the main steps of the GA. The chromosome coding is similar to the coding employed in [19]. As each image was partitioned into m regions, each chromosome (C) contains m genes $(G_1, G_2, G_3, \ldots, G_m)$. Moreover, each gene (G_i) contains a vector of four weights, with the first quantifying the region importance and the other ones quantifying the importance of the color, shape and texture descriptors, respectively. We have tested $m = 4$, $m = 9$, $m = 16$ and $m = 25$. The best result obtained from these empirical tests was $m = 16$, which was defined as default.

Algorithm 1 GA-based RF algorithm

```
Require: Query image ($q$), user's feedback, feature vectors, GA's parameters.
Ensure: Optimized set of weights for image feature vectors ($\omega_F$) and image regions ($\omega_R$).
1: Generate a population ($P$) of random individuals ($\mathcal{C}$) according to the chromosome coding ($C$);
2: Evaluate each individual $\mathcal{C}$ of $P$ according to some fitness function given by a REF;
3: Select individuals by the roulette-wheel method until the mating pool is completed;
4: Apply uniform crossover and uniform mutation on the selected individuals;
5: Select the best individuals among parents and offsprings for the next generation;
6: While the number of generation is not exhausted, return to step 2.
7: Return the weights set $\{\omega_F, \omega_R\}$ coded by the fittest individual.
```

Ranking evaluation functions

We compared ten REFs being two not based on the utility theory (nUTB-REF) and eight based on the utility theory (UTB-REF). Utility theory-based fitness functions (UTB-REFs) are based on the utility concept, where the score value of a relevant element in the ranking is usually inversely proportional to its position. That is, the higher the rank of a relevant element, the higher its utility. Non-utility theory-based fitness functions (nUTB-REFs) are REFs that do not strictly follow the utility concept.

A REF plays the role of the GA fitness function, and it is applied as described in Algorithm 2. First, the image similarities (Equation 1) between the query image and each image in the data set are computed by employing the weights coded by the individual $\mathcal{C}$. Then, the images are sorted according to the similarity values which make up a ranking. Finally, a ranking evaluation function is applied to the ranking to obtain the fitness value. In the following, we describe the ranking evaluation functions analyzed, grouping them into two categories: nUTB-REF and UTB-REF. Fitness$(q, \mathcal{C})$ denotes the fitness value of the individual $\mathcal{C}$ for the query q, I represents the image data set, $|I|$ denotes the cardinality of I, D represents the set of images known to be relevant to a query q, $|D|$ denotes the cardinality of D and pos(i) returns the position (rank) of the image i in the ranking.

Algorithm 2 Fitness function employment

```
Require: Image query ($q$), user's feedback, feature vectors, individual $\mathcal{C}$.
Ensure: Fitness value.
1: For each image $i$ in the data set:
2:     Compute the image similarity between $q$ and $i$ (Equation 1);
3: Sort the images according to the similarity values;
4: Compute the fitness value of the ranking employing a ranking evaluation function;
```

Non-utility theory-based fitness functions

The non-utility theory-based fitness functions are as follows:

- *Fitness function F1.* This fitness function is given by the R-precision measure, which is a well-known REF used to evaluate information retrieval effectiveness:

$$\mathrm{F1}(q, \mathcal{C}) = R\text{-precision}(q, \mathcal{C}) = \frac{\text{Number of relevant images retrieved}}{n_R}, \qquad (2)$$

  where n_R is the number of elements considered in the query answer.
- *Fitness function F2.* This function is based on an analysis of the numbers of *true positive* (Rr - relevant and retrieved items), *false positive* (Rn - retrieved but non-relevant items) and *false negative* (Nr - non-retrieved relevant items):

$$\mathrm{F2}(q, \mathcal{C}) = (2|D|) + \mathrm{Rr} - \mathrm{Rn} - \mathrm{Nr}. \qquad (3)$$

The fitness function F1 was employed in Stejić et al. models [19], and $F2$ was proposed in [18].

Utility theory-based fitness functions

Utility theory-based fitness functions (UTB-FFs) are fitness functions based on UTB-REFs. We analyzed eight UTB-FFs ($F3$ to $F10$) defined as follows:

- *Fitness function F3*

$$\mathrm{F3}(q,\mathcal{C}) = \frac{1}{|D|}\sum_{\forall i\in D}\left(\sum_{j=\mathrm{pos}(i)}^{|I|}\frac{1}{j}\right) \tag{4}$$

- *Fitness function F4*

$$\mathrm{F4}(q,\mathcal{C}) = \sum_{\forall i\in D}\left(\frac{1}{A}\left(\frac{(A-1)}{A}\right)^{(\mathrm{pos}(i)-1)}\right), \tag{5}$$

  where A is a user-defined parameter with values larger than or equal to 2.

- *Fitness function F5*

$$\mathrm{F5}(q,\mathcal{C}) = \frac{\text{Accuracy value}(q,C)}{\sum_{j=1}^{|D|}\frac{1}{j}}, \tag{6}$$

  where

$$\text{Accuracy value}(q,\mathcal{C}) = \sum_{\forall i\in D}\frac{1}{\mathrm{pos}(i)} \tag{7}$$

- *Fitness function F6*

$$\mathrm{F6}(q,\mathcal{C}) = \sum_{\forall i\in D} k_1 ln^{-1}(\mathrm{pos}(i)+k_2), \tag{8}$$

  where k_1 and k_2 are user-defined parameters.

- *Fitness function F7*

$$\mathrm{F7}(q,\mathcal{C}) = \sum_{\forall i\in D} k_3 \log_{10}(|I|/\mathrm{pos}(i)), \tag{9}$$

  where k_3 is a user-defined parameter.

- *Fitness function F8*

$$\mathrm{F8}(q,\mathcal{C}) = \sum_{\forall i\in D} k_4^{-1}(e^{-k_5 \ln(\mathrm{pos}(i))+k_6} - k_7), \tag{10}$$

  where k_4, k_5, k_6 and k_7 are user-defined parameters.

- *Fitness function F9*

$$\mathrm{F9}(q,\mathcal{C}) = \sum_{\forall i\in D} k_8 k_9^{\mathrm{pos}(i)}, \tag{11}$$

  where k_8 and k_9 are user-defined parameters.

- *Fitness function F10*

$$\mathrm{F10}(q,\mathcal{C}) = \frac{\sum_{\forall i\in D}\left(\frac{\sum_{j=1}^{\mathrm{pos}(i)} r\left(\arg ii{:}\mathrm{pos}(ii){=}{=}j\right)}{\mathrm{pos}(i)}\right)}{|D|}, \tag{12}$$

  where $r\big(\arg ii : \mathrm{pos}(ii) == j\big)$ returns 1 if the image ii in the jth position of the ranking is relevant, otherwise it returns 0.

Fitness functions F3 and F4 were used in [17] for the learning of weights, which were structured according to the vectorial space model, in the context of textual information retrieval. The fitness function F5 was proposed in [18], and the functions F6 to F10 are used in [13] and [5] for GP-based ranking function discovery to improve textual information retrieval and CBIR tasks, respectively.

Data sets

We evaluated the REFs for the weighting of features in image retrieval on five public domain image data sets, varying from hundreds to ten thousand images. The image data sets employed are summarized in Table 1.

Results and discussion

Previous studies on rank learning methods [5,13,17,20] show that, in general, UTB-REFs lead to more precise information retrieval results than nUTB-REFs. Moreover, these studies show that the UTB-REFs' design by itself significantly affects the information retrieval results. In our study, we performed a systematic investigation of REFs for descriptor/region weighting in image retrieval using the successful model WLSP-C± (Equation 1). Considering the comparison of REFs, although our results were in line with those reported in the literature, we found better results with the UTB-REF F5, which has not been investigated in other research studies.

As can be seen in Figure 2, the UTB-REF F5 was on average more precise than the other REFs, when considering low recall rates. For all data sets, the images belonging to the same category of the query image were considered as relevant, while the remaining images were considered irrelevant. The result shown in Figure 2 has high significance, since users largely emphasizes the analysis on the best ranked items. Therefore, the closer to the top ranking the relevant items appear, the better the result. As REFs play a key role in ranking optimization and given the importance of high precision in top-k ranking for several applications, it is conceived that the UTB-REF F5 could be effectively applied in other researches focused on ranking

Table 1 Data sets used in the experiments

Data set name	Number of images	Number of classes	Images per class
Vistex-167 [21]	167	19	2 to 20
Corel-1000 [22]	1,000	10	100
DB-10000 [18]	10,000	-	-
Scenes-1044 [23]	1,044	25	328 to 360
Caltech101-8872 [24]	8,872	47	31 to 800

DB-10000 data set contains 1,000 images imported from Corel-1000 data set and 9,000 images which were not pre-classified.

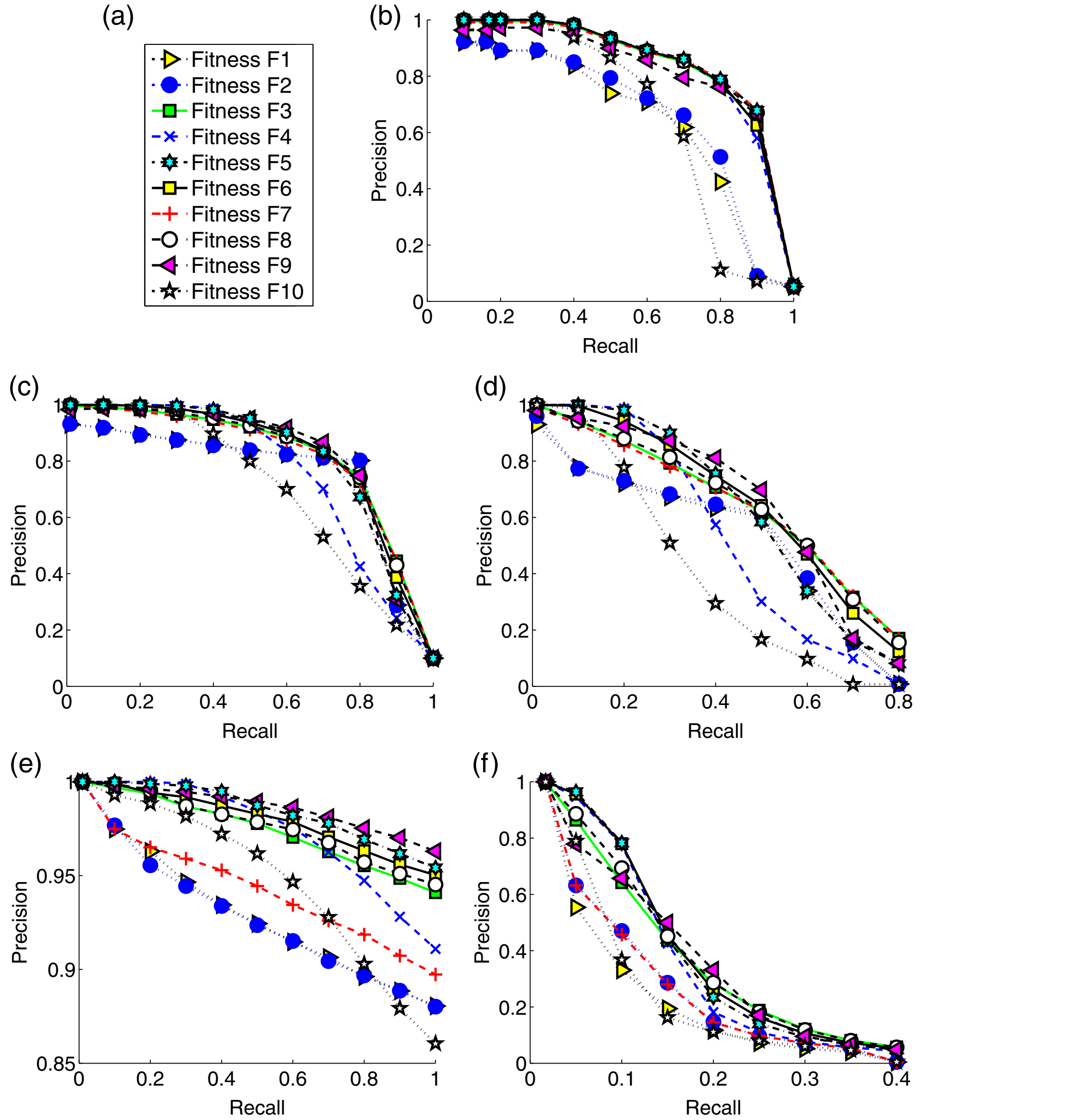

Figure 2 Average P&R graphs for all evaluation (fitness) functions. Analyzed on **(b)** Vistex-167 data set, **(c)** Corel-1000 dataset, **(d)** DB-10000 data set, **(e)** Scenes-1044 data set and **(f)** Caltech101-8872 data set. The legend for the graphs is given inside **(a)**. The parameters used were the same as those found in studies from the literature: $k_1 = 6$, $k_2 = 1.2$, $k_3 = 2$, $k_4 = 3.65$, $k_5 = 0.1$, $k_6 = 4$, $k_7 = 27.32$, $k_8 = 7$, $k_9 = 0.982$ and $A = 10$. P&R graphs for Vistex-167 and Corel-1000 have been obtained using all the images of the data sets as queries. P&R graphs for DB-10000 data set have been obtained using 1,000 query images, originating from Corel-1000 data set.

optimization. Moreover, the application of F5 is straightforward since it has no parameter adjustment. Table 2 shows the area under the precision recall curve referred to in Figure 2, bounded at 25%, 50% and 75% of recall. One observes that fitness F5 only loses out to the others on the BD-10000 in 75% of recall, which confirms the superiority of fitness F5.

By analyzing the REF behaviour, we realize that the superiority of F5 is due to the highest relative importance that it attaches to the top positions of the ranking. According to the authors belief, this corresponds to a near-optima utility function because when performing a query the user wants relevant documents in the first positions of the ranking. As an example, let us take a hypothetical situation of two rankings with n retrieved images: in the first ranking, we have a relevant image in the first position and another relevant image in the last position with other positions occupied by non-relevant images; in the second ranking, we have two relevant images in the second and third positions with the other retrieved images being non-relevant. In general, from a user's point of view, having a relevant image in the first position is more important

Table 2 Area under the precision recall curve referred to in Figure 2, bounded at 25%, 50% and 75% of recall

	Recall (%)	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
Data set Vistex-167	25	0.228	0.229	0.249	0.250	0.250	0.250	0.248	0.250	0.242	0.250
	50	0.436	0.440	0.494	0.496	0.496	0.496	0.492	0.496	0.480	0.484
	75	0.603	0.616	0.713	0.716	0.717	0.716	0.711	0.716	0.691	0.654
Data set Corel-1000	25	0.219	0.219	0.237	0.240	0.240	0.240	0.236	0.238	0.237	0.240
	50	0.435	0.434	0.475	0.486	0.486	0.482	0.472	0.477	0.480	0.468
	75	0.640	0.640	0.693	0.682	0.706	0.702	0.687	0.696	0.705	0.630
Data set DB-10000	25	0.185	0.186	0.220	0.238	0.238	0.234	0.218	0.222	0.227	0.214
	50	0.346	0.350	0.403	0.398	0.434	0.428	0.399	0.408	0.431	0.306
	75	0.429	0.437	0.514	0.439	0.512	0.531	0.511	0.519	0.528	0.326
Data set Scenes-1044	25	0.233	0.233	0.239	0.240	0.240	0.239	0.234	0.239	0.239	0.238
	50	0.468	0.467	0.485	0.488	0.489	0.486	0.472	0.485	0.487	0.482
	75	0.696	0.695	0.727	0.731	0.734	0.730	0.705	0.728	0.733	0.717
Data set Caltech101-8872	25	0.073	0.090	0.125	0.128	0.135	0.133	0.089	0.129	0.128	0.083
	50	0.080	0.100	0.143	0.140	0.151	0.147	0.099	0.147	0.143	0.091
	75	0.081	0.101	0.145	0.141	0.152	0.148	0.100	0.148	0.144	0.092

than having the first position occupied by a non-relevant element followed by two relevant images. F5 is in accordance to this behaviour for all values of n. Moreover, F5 is the only function from the REFs analysed which is in accordance to this behaviour for $n > 30$. Table 3 shows the scores assigned to the hypothetical rankings for $n = 31$.

Also, in reference to Figure 2, we found that the P&R graphs obtained using UTB-REFs (F3–F10) are noticeably different from those obtained using nUTB-REFs (F1 and F2). One easily notes that, in general, the UTB-REFs produced substantially higher precision values than the nUTB-REFs (F1 and F2), when considering low recall rates. This is a very important aspect that has not been discussed by other researchers. Utility theory-based evaluation functions enable these sort of results, due to the fact that they allow for the appropriate modeling of the user requirements in regard to ranking quality.

Another important issue observed in the analyses performed is that the global computational time spent when using a proper UTB-REF is significantly lower than when using a well-known nUTB-REF, such as the *R*-precision measure. Once all the UTB-REFs investigated take a similar computational procedure, one can choose one of them when analyzing computational time without loss of generality. We chose the UTB-REF F5 and compared it against the nUTB-REF F1. We evaluated the number of generations and the computational time spent by the GA during the RF process. As the maximum feasible fitness value is sometimes not achieved by the GA, it was considered that individuals could evolve up to 350 generations. For assessment, we carried out 100 queries in the *DB-10000* data set by random selection of 10% of the images for each category coming from the *Corel-1000* data set, and we reported the average values obtained. The system was fed back with the first ten relevant images of the initial ranking for both methods. One can see in Table 4 that the computational time when using the UTB-REF F5 was on average 2.8 times faster than when using the nUTB-REF F1. Also, one can see that when using the UTB-REF F5, the GA spent on average 3.4 less generation than when using F1. In summary, Table 4 shows that in spite of UTB-REF being a little more expensive computationally, the GA-based RF process needed a significant smaller number of generations to obtain results of greater superiority than when using the nUTB-REF

Table 3 Scores assigned for two hypothetical rankings with 31 retrieved images

Ranking	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
First	0.065	−23	2.03	0.104	0.688	9.338	2.982	10.599	10.86	0.532
Second	0.667	5	2.777	0.171	0.556	9.339	4.409	12.389	13.379	0.583

The first ranking is composed of a relevant image in the first position and another relevant image in the last position with other positions occupied by non-relevant images. The second ranking is composed of two relevant images in the second and third positions with the other retrieved images being non-relevant.

Table 4 Average number of GA generations and computational time (in seconds) spent

Data set category	Fitness F1 generations	Time	Fitness F5 generations	Time
Africa	324	147.064	82	45.422
Beach	303	137.532	51	28.275
Buildings	174	78.979	39	21.622
Buses	95	43.121	25	13.861
Dinosaurs	242	109.844	47	26.058
Elephants	335	152.057	98	54.333
Flowers	17	7.716	15	8.316
Food	135	61.277	87	48.235
Horses	21	9.532	16	8.871
Mountains	331	150.241	117	64.867
Average	198	89.872	58	32.156

F1. All experiments were executed in a Windows 7 64-bit OS using an Intel Core 2 Duo 2.2-GHz processor with 4-GB RAM. The prototype was implemented in ANSI C.

We also found, for all data sets, that the GA-based RF technique produced P&R graph results far superior than a similar RF technique employing multistart (MS) search instead of GA search. This result is shown in Figure 3. The number of random solutions of MS search was set to the same number of fitness evaluations performed by the GA in all the comparative experiments carried out, i.e. $S_p(1 + p_c(n_G - 1)$, where S_p is the population size, p_c is the crossover rate and n_G is the number of generations of the GA search. For both these search techniques, GA and MS search, the fitness function F5 was employed as the evaluation criterion. MS search may be naturally compared with GA search, since both employ random mechanisms. This result shows the strength of GA for this sort of optimization.

Finally, we provided a study for the computational complexity of the RF technique, and we found that it is linear regarding the number of images in the data set. We analyzed the number of similarity operations (Equation 1) computed by the fitness function during the evolutive process, as the similarity calculus is the most expensive operation in the RF process.

In Algorithm 1, step 1 has complexity $O(1)$, as it does not depend on the number of images in the data set. In step 2, the fitness score for each individual $\mathcal{C}$ is computed employing Algorithm 2. Analyzing the Algorithm 2, it is trivial to find out that the image similarity operation (step 2) takes time $O(n)$, where n is the number of images in the data set. Step 3 is $O(n \log n)$ – time for sorting the similarity values of n images. However, the image similarity operation takes significantly larger computational time than value comparisons and exchanges of sorting algorithms, even for considered unthinkably large image data sets today (containing several million or more elements). Thus, we consider as the main operation of Algorithm 2, i.e. the time unit, the number of operations performed by the similarity query process that increases in $O(n)$.

Returning to Algorithm 1, any of the steps 3 to 7 has complexity $O(1)$ for the same reason as step 1. In summary, as the fitness function is applied a constant number of times, depending on the population size, generation number and crossover rate, the GA-based RF algorithm is $O(1)O(n)$, i.e., linear. It is important to remember that the constant term $O(1)$ can be significantly high, depending on the GA parameters. However, the fitness operations can be performed in a parallel fashion in each GA generation.

Conclusions

As known from many research studies, the objective function plays a crucial role in ranking optimization. In this study, we present an up-to-date investigation of ranking evaluation functions (REFs), a special class of objective function employed in rank learning methods aimed at providing precise information retrieval. Using a GA-based RF method as a rank learning mechanism for image retrieval, we analyzed ten REFs, which includes the most successful REFs employed in previous studies regarding comparison of REFs adding some functions not investigated.

We performed an analysis of precision-recall curves in five real-world image data sets. Although our results were in line with those reported in the literature, showing that the REF design has a decisive hole in rank learning, we found that the UTB-REF named here F5, which is not included in previous studies that compared REFs, provided better results than the recommended REFs. Additionally, the computation of F5 does not require any parameter, to the contrary of previously recommended REFs. Also, we found that UTB-REF is the most appropriate class of REF for top-ranking optimization. Another important issue noticed is that the time spent in the ranking optimization process when using a proper UTB-REF, such as F5, is significantly lower than when using a well-known nUTB-REF, such as the *R*-precision measure. Showing the strength of GA search for the optimization task, we compared and found that GA significantly overcame multistart (MS) search. This result shows that GA search is effective for learning weights through RF aiming at optimizing image retrieval results.

Our results added to those from the literature, showing a categorization and a systematic analysis of REFs and

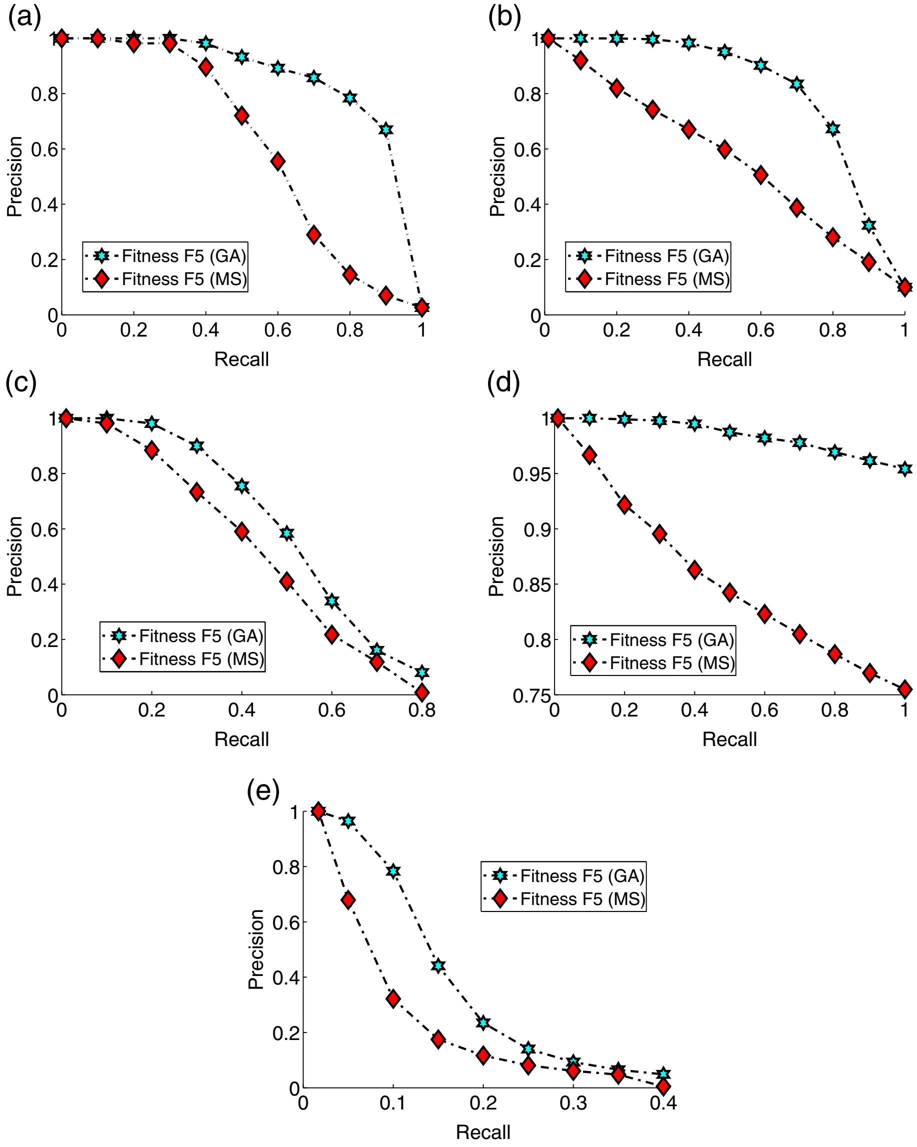

Figure 3 Precision-recall graphs for GA and MS search. (a) Vistex-167 data set, **(b)** Corel-1000 data set, **(c)** DB-10000 data set, **(d)** Scenes-1044 data set and **(e)** Caltech101-8872 data set. The candidate solutions of MS search were represented in the same way of the GA candidate solutions.

confirming that the REF design plays a key role in rank learning. To the best of our knowledge, this is the first study carried out to investigate the importance of REFs in feature weighting for CBIR tasks.

As REFs play a key role in many ranking optimization tasks, our results indicate that REF F5 could be effectively applied in other contexts and applications focused on ranking optimization, such as recommender systems: the idea here is to provide recommendations sorted according to their expected utility, such as user rating and/or similarity according to the user's interests. Also, we put together and compared a broad set of REFs that can be used for future research in the ranking optimization field.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

All authors have contributed to the different methodological and experimental aspects of the research. All authors read and approved the final manuscript.

Acknowledgements

We thank CNPq, CAPES and FAPESP for the financial support.

Author details

[1]Federal University of Goiás, Catalão, 75604-020 Goias, Brazil. [2]University of São Paulo, São Carlos, 13566-590 São Paulo, Brazil. [3]Federal University of Uberlândia, Uberlândia, 38400-902 Minas Gerais, Brazil.

Received: 19 February 2013 Accepted: 2 December 2013
Published: 6 March 2014

References

1. Adomavicius G (2012) Improving aggregate recommendation diversity using ranking-based techniques. IEEE Trans Knowl Data Eng 24(5):896–911
2. Liu TY (2009) Learning to rank for information retrieval. Foundations Trends Inf Retrieval 3(3):225–231
3. Pedronette D, Torres R (2012) Exploiting contextual information for image re-ranking and rank aggregation. Int J Multimedia Inf Retrieval 1:1–14
4. Qin T, Liu TY, Xu J, Li H (2010) LETOR: a benchmark collection for research on learning to rank for information retrieval. Inf Retrieval 13(4):346–374
5. Torres RS, Falcão AX, Gonçalves MA, Papa JP, Zang B, Fan W, Fox EA (2009) A genetic programming framework for content-based image retrieval. Pattern Recognit 42(2):283–292
6. Vargas S, Castells P (2011) Rank and relevance in novelty and diversity metrics for recommender systems. In: Proceedings of the fifth ACM conference on recommender systems. Chicago, 23–27 October 2011, pp109–116
7. Ah-Pine J (2011) On data fusion in information retrieval using different aggregation operators. Web Intell Agent Syst 9:43–55
8. Ailon N (2008) Aggregation of partial rankings, p-ratings and top-m lists. Algorithmica 57(2):284–300
9. Lin S (2010) Rank aggregation methods. Wiley Interdiscip Rev: Comput Stat 2(5):555–570
10. Nuray R, Can F (2006) Automatic ranking of information retrieval systems using data fusion. Inf Process Manag 42(3):595–614
11. Drosou M, Pitoura E (2010) Search result diversification. ACM SIGMOD Rec 39(1):41–47
12. Santos R, Macdonald C, Ounis I (2010) Exploiting query reformulations for web search result diversification. In: Proceedings of the 19th international conference on World Wide Web, WWW '10, Raleigh, 26–30 April 2010, pp881–890
13. Fan W, Fox EA, Pathak P, Wu H (2004) The effects of fitness functions on genetic programming-based ranking discovery for web search. J Am Soc Inf Sci Technol 55(7):628–636
14. Ferreira C, Santos J, Torres RS, Gonçalves M, Rezende R, Fan W (2011) Relevance feedback based on genetic programming for image retrieval. Pattern Recognit Lett 32(1):27–37
15. López-Pujalte C, Guerrero-Bote VP, De Moya-Anegón F (2003) Genetic algorithms in relevance feedback: a second test and new contributions. Inf Process Manag 39(5):669–687
16. López-Pujalte C, Guerrero Bote VP, Moya-Anegón F (2002) A test of genetic algorithms in relevance feedback. Inf Process Manag 38(6):793–805
17. López-Pujalte C, Guerrero-Bote VP, Moya-Anegón F (2003) Order-based fitness functions for genetic algorithms applied to relevance feedback. J Am Soc Inf Sci 54(2):152–160
18. Silva SF, Barcelos CAZ, Batista MA (2007) Adaptive image retrieval through the use of a genetic algorithm. In: Proceedings of IEEE international conference on tools with artificial intelligence (ICTAI), Patras, 29–31 October 2007, pp557–564
19. Stejić Z, Takama Y, Hirota K (2003) Genetic algorithms for a family of image similarity models incorporated in the relevance feedback mechanism. Appl Soft Comput 2:306–327
20. Fan W, Pathak P, Zhou M (2009) Genetic-based approaches in ranking function discovery and optimization in information retrieval—a framework. Decis Support Syst 47:398–407
21. Massachusetts Institute of Technology Media Laboratory (2005) Vistex database. http://vismod.media.mit.edu/pub/VisTex/. Last accessed on 06 Feb 2014
22. James Z. Wang's Research Group. Corel database (2004) Corel Corporation, Corel Gallery 3.0. http://wang.ist.psu.edu/~jwang/test1.tar. Last accessed on 06 Feb 2014
23. Vision Lab. in Computer Science Department (2004) 13 scene categories database. http://vision.stanford.edu/Datasets/SceneClass13.rar. Last accessed on 06 Feb 2014
24. Fei-Fei L, Fergus R, Perona P (2004) Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories In: IEEE CVPR 2004 workshop on generative-model based Vision (IEEE, Piscataway, 2004)

doi:10.1186/1678-4804-20-7
Cite this article as: Silva *et al.*: **Findings on ranking evaluation functions for feature weighting in image retrieval.** *Journal of the Brazilian Computer Society* 2014 **20**:7.