

# Rank aggregation using ant colony approach for metasearch

Parneet Kaur<sup>1,2</sup> · Manpreet Singh<sup>3</sup> · Gurpreet Singh Josan<sup>4</sup> · Sukhwinder Singh Dhillon<sup>5</sup>

Published online: 22 July 2017  
© Springer-Verlag GmbH Germany 2017

**Abstract** Metasearch engines provide a plethora of information to the user through World Wide Web. They are the prominent sources of query-based search and centralized human–world interactions. Metasearch engine shows a list of Web sites to a particular query as per the rank assigned to a web link. The effectiveness of metasearch engine is also examined on the basis of ranks assigned to Web sites for a particular query. Assigning top rank to a web link with most relevant information pertaining to a query by the search engine is formulated as research problem. Here, we have formulated the rank aggregation optimization problem by using metaheuristic approach. Search engines are facing widely two problems such as biasing of search solutions and giving irrelevant rank to similar kind of documents. Both these problems can be overcome by applying an effective rank aggregation technique for combining the search results from various search engines. This paper presents a metaheuristic approach to optimize Spearman’s footrule and Kendall-tau distance measures which are used to compare ranking methods. The performance of proposed ant colony-based strategy

is compared with GA technique and is validated through experimental results for real-world queries. Likewise, Precision, Recall and F-Measure-based performance metrics are employed to test the effectiveness of various metasearch engines.

**Keywords** Metasearch · Rank aggregation · Genetic algorithm · Ant colony algorithm

## 1 Introduction

In the present scenario, real and interesting datasets are available in abundant amount worldwide for any discipline such as sports, politics, education, psychology, stock market. But searching of useful information is the tedious and complex task. Metasearch engines provide lists of different Web sites to a user query containing its keywords. These lists merely match the user satisfaction level. Hence, rank aggregation of several lists into one “super” list is one of the best solutions for getting better results. The various rank aggregation techniques were employed in literature such as score-based methods, positional methods, probabilistic methods, learning techniques and soft computing techniques (Beg and Ahmad 2003, 2004; Beg et al. 2016; Dwork et al. 2001; Renda and Straccia 2003; Montague et al. 2001). In order to investigate the performance of rank aggregation techniques, two distance measures such as Spearman’s footrule and Kendall-tau are used. The distance value of final aggregated list is calculated among other input lists. Hence, a normalized aggregated distance is obtained and top rank is assigned to a Web site having minimum distance value. The minimization of this normalized aggregated distance is known as optimal aggregation which is an NP-hard problem. A common rank aggregation technique was the Borda’s method developed in 1770s by

Communicated by V. Loia.

✉ Parneet Kaur  
kaur.parneet@gmail.com

<sup>1</sup> Department of Computer Science and Engineering, I.K.G Punjab Technical University, Jalandhar, Punjab, India

<sup>2</sup> Department of Computer Science and Engineering, ACE & AR Devsthali, Ambala, Haryana, India

<sup>3</sup> Department of Computer Science and Engineering, GNDEC, Ludhiana, Punjab, India

<sup>4</sup> Department of IT and Computer Science and Engineering, Punjabi University, Patiala, Punjab, India

<sup>5</sup> Department of Electronics and Communication Engineering, ACE & AR Devsthali, Ambala, Haryana, India

Borda (1781). Although this method was applied on the political dataset, but later on it was investigated with several new problems like sports, psychology, stock market as application areas (Langville and Meyer 2012). In 1951, Arrow has proved his impossibility theorem, which describes the limitations inherent in the voting system. Shimura (1973) proposed a fuzzy logic technique for ranking of objects. Dubois and Prade's (DP) (Ross 1997) technique was adopted for rank aggregation which was based on the on the fuzzy ordering of elements. In this technique apart from mean position of the document, averages over the results were also used. This approach was missing in the Borda's method. Entropy technique by Ross (1997) was employed for ranking large datasets with lot of variation and complexity by using the concept of threshold for obtaining position of documents in a given range.

In 1998, paper by Larry and Brin (Page and Brin 1998) "Anatomy of a Search Engine" opened new horizons in the field of ranking and rating by designing Google search engines which was based on Markov chains to rank web pages. The Google search engine was based on the PageRank algorithm which helped in calculating the relevancy of new pages in comparison with other existing pages. PageRank (Page and Brin 1998) and Kleinberg's HITS (Kleinberg 1999) algorithms are currently used for measuring the importance of a document on web. Dwork et al. (2001) considered the problem of rank aggregation in context of web. They studied rank aggregation technique to overcome the problem of *spam fighting*. Montague et al. (2001) came up with various methods of metasearch. His work was focused on the normalization score values which were used to decompose the metasearch concept.

Beg and Ahmad (2003) employed genetic algorithm (GA) for optimization of Spearman footrule distance for partial lists known as partial footrule optimal aggregation (PFOA). Beg and Ahmad (2004) proposed three strategies which were compared with classical fuzzy technique and Borda's method for web applications. Yan et al. (2011) improved the quality of PageRank algorithm by introducing a new algorithm based on genetic algorithm known as a genetic PageRank (GPRA). The GPRA algorithm was based on the topic-sensitive idea which is divided into two phases: a collection of themes related to the calculation of PageRank vectors and online inquiries to determine the context of theme. The problem of refinement of page ranking was considered by Laughlin et al. (2011) using fuzzy set and logic. Akritidis et al. (2011) highlighted the effective rank aggregation techniques for metasearch. In his paper QuadRank, a new rank aggregation method was introduced which consider the query terms, the collected results and the data correlated with each of these results related to title, textual, snippet, URLs and individual rankings, etc. Likewise, the concept of feature selection was employed by Waad et al. (2013) in the area of rank aggrega-

tion. Rank aggregation problem was investigated (Pihur et al. 2014) on R tool for computing cross-entropy. The concept of multiple objective GA-based rank aggregation techniques was addressed by Kaur et al. (2015). The Mean and Stuart methods were applied to ranking of different queries. Beg et al. (2016) covered the rank aggregation problem in metasearch by modified Shimura method with addition of OWA operator. Kaur et al. (2017) had implemented the genetic algorithm by optimizing the distance measures for various URLs obtained from different search engines using rank aggregation techniques. Genetic-based Kendall-tau as (GKTu) and Spearman's foot rule as (GSFD) distance measures were proposed to derive optimized distance measure values. However, the work reported by Beg and Kaur provide better solution to ranking problem, but it is having drawback of large execution time. Napoles et al. (2015) presented a model based on ant colony for partial ranking using the Kemeny approach.

In the past few years, more advanced nature inspired optimization techniques were proposed, such as ant colony, PSO and bee colony for the solutions to dynamic and mathematical problems. These techniques are yet to be applied for ranking of Web sites. Highly motivated from the different ranking techniques and need to provide better strategy for rank aggregation having less execution time are taken as problem objectives in this work.

In this paper, a bio-inspired ant colony-based rank aggregation methodology is proposed. Simulation solutions are received for five aggregation methods, such as Borda, Markov chain, scaled footrule, PageRank and mean-by-variance. At a later point, the execution of the suggested scheme is compared with traditional genetic algorithm on the basis of implementation time and minimized distance value. At last, three main performance metrics such as Precision, Recall and *F*-Measure are also discussed for checking the efficiency of various search engines.

## 2 Rank aggregation methods used for metasearch

The area of rank aggregation is divided into two categories, i.e., unsupervised and supervised learning approach. In case of unsupervised approach, no training dataset is required for deriving the results like Borda count (Borda 1781; Saari 2000), median rank aggregation by Dwork et al. (2001), genetic and fuzzy-based rank aggregation (Beg and Ahmad 2002), whereas the supervised techniques utilizes a dataset.

### 2.1 Kemeny optimal aggregation method

Kemeny optimal aggregation (Kemeny and Snell 1962) is one of the desirable methods used for ranking problems. Given a set of total rankings  $\sigma^1, \dots, \sigma^j, \dots, \sigma^N$  on the set of items  $Z$ , the goal is to find the order of  $Z$ ,  $\sigma$ . Suppose that the

input rankings  $\sigma^j$  are noisy versions of  $\sigma$ , which is obtained by swapping two elements of  $\sigma$  with a probability  $r < 0.5$  given as.

$$\sigma^* = \arg \min_{\sigma} \frac{1}{N} \sum_{j=1}^N K(\sigma, \sigma^j) \quad (1)$$

The estimate  $\sigma^*$  is referred as to Kemeny optimal aggregation. [Dwork et al. \(2001\)](#) in his work had shown that the computation of Kemeny optimal aggregation is a NP-hard problem even when the number of rankings is four.

## 2.2 Borda's method

Borda's method ([Borda 1781](#)) is one of the classical method based on the position oriented score. The scores are assigned as per the position by which a candidate appears within each voter's ranked list of preferences. The score value of all candidates is used for sorting. In this method, a score is assigned first, suppose  $S_i(c)$  = the number of candidates ranked below  $c$  in  $\alpha_i$  is the list and  $\alpha_1, \alpha_2, \dots, \alpha_k$  is the set of full list. For each candidate  $c \in D$ , the total Borda score is expressed as:

$$\sum_{i=1}^k S_i(c) \quad (2)$$

The candidates are sorted in decreasing order of the Borda score.

## 2.3 Markov chain method

This method of ranking designed by Markov in 1906 and its counter parts were proposed by [Dwork et al. \(2001\)](#). Markov chains for a system involves a set of states having different items. A nonnegative stochastic matrix  $M$  of  $n \times n$  item is formed. The method progresses initially by moving from a particular state to other state. The  $M_{ij}$  value indicates the state  $i$  to state  $j$  movement. The Markov chain comprised of probability distributed matrix formed by the product of  $M$  and current state distribution. The start state of distribution is indicated by  $x$  on  $N$ . The process follows a number of iterations continue until distribution reaches to a steady-state value which is known as stationary distribution. The stationary distribution is given by the principal of left eigenvector  $y$  of  $M$ , i.e.,  $yM = \gamma y$ . All entries in  $y$  are known as natural ordering on  $S$  which is called as Markov chain ordering of  $M$  as MC1, MC2, MC3 and MC4. These chains can be applied on different applications as per their requirements.

## 2.4 PageRank method

[Page and Brin \(1998\)](#) developed the PageRank algorithm at Stanford University. This algorithm had become the back

bone of Google search engine along with several other search engines. The page rank methods calculate the relevancy of pages in comparison with other existing web pages. Other parameters like title, tag, anchor tag, keywords are also used to calculate the overall rank of page under PageRank.

Final results are arranged in descending order using following relation as.

$$PR(u) = (1 - d) + d \sum_{v \in B(u)} \frac{PR(v)}{N_v} \quad (3)$$

where  $PR(u)$  denotes the page rank for page ( $u$ ) and  $N_v$  denotes the number of out-links of page ( $v$ ).  $d$  is a dampening factor, and its value is 0.85. The value of  $d$  can vary as per user's requirement and following the direct links and  $(1 - d)$  as the page rank distribution from non-directly linked pages.

## 2.5 HITS algorithm

HITS algorithm ([Kleinberg 1999](#)) is based on the concept of authority and hub instead of in-links and out-links concept for allotting rank to the pages. The number of edges among hubs and authority was taken for calculations. Weights were assigned to hubs and authority in order to modify the existing results. The algorithm works in the form of iterations for calculation of ranks. As first iteration starts, the weight assigned to a node is computed as:

$$a_i = \sum_{j \in B(i)} h_j \quad \text{and} \quad h_j = \sum_{i \in F(j)} a_i \quad (4)$$

where  $a_i$  is the authority update connected to  $i$  and  $h_j$  is the hub update connected to  $j$ .

## 3 Rank aggregation-based distance measures

Rank aggregation method can be defined as generating a new list of elements which is closest to a set of given lists. The performance of a rank aggregation method is measured by two distance measures, namely Kendall-tau distance and Spearman's footrule distance. These measures were first reported by [Diaconis and Graham \(1977\)](#) and [Diaconis \(1988\)](#).

### 3.1 Kendall-tau distance measure

This method is widely used to calculate the correlation between two rank lists. Kendall-tau distance in [Kendall \(1938\)](#) counts the number of inversion in the order of pair of candidates between two rank lists. The number of pairwise disagreements is counted by the Kendall-taudistance

between two full lists. The distance variables taken for two full lists  $k$  and  $l$  are denoted by  $\sigma$  and  $\tau$  formulated as:

$$K(\sigma, \tau) = |\{(k, l); k < l, \sigma(k) < \sigma(l) \text{ but } \tau(k) > \tau(l)\}| \quad (5)$$

The normalized value for Kendall-tau distance is derived by dividing with greatest possible coefficient value of 0.5.

### 3.2 Spearman's footrule distance measure

The Spearman's footrule distance (Diaconis and Graham 1977) is defined as absolute distance between positions of candidates in two ranking lists. Suppose two document lists  $l1 : (d1, d2, d3, d4)$  and  $l2 : (d4, d3, d2, d1)$  are given. Spearman's footrule distance  $d$  is calculated as the sum of absolute difference between positions of documents, e.g., position of  $d1$  in  $l1$  is 1st and  $l2$  is 4th, and therefore, the contribution of document  $d1$  to distance is  $|1 - 4|$ . Hence, total distance is calculated as.

$$d = |1 - 4| + |2 - 3| + |3 - 2| + |4 - 1| \\ d = 8$$

The Spearman's footrule distance calculates the sum of all elements, i.e.,  $k$  for a given set, suppose 'A' is generated.

In this work, minimization of Kendall-tau and Spearman's footrule distance measures is formulated as objective functions:

$$\text{Obj\_function} = \min\{K(\sigma, \tau), F(\sigma, \tau)\} \quad (7)$$

### 4 Rank aggregation by genetic algorithms approach

Genetic algorithm (GA) was developed by Holland (1975) which forms the strong foundation for evolutionary computing (Yang 2010, 2014). The whole process of GA is basically divided into three steps which are crossover, mutation and selection. The use of GA for NP-hard problem was firstly applied by Goldberg (1989) in search and machine learning area. The general procedure for GA is to encode the objective or cost. Secondly, define the fitness function or selection criterion, and then create a population of individuals. Then, an iteration or evolution cycle is carried out by evaluating the fitness of all the individuals in the population. Also with these cycles, new population are created by using the older one. Lastly decode the results in order to obtain the optimized value (Yang 2010). The pseudo-code for GA optimized distance measures for rank aggregation is given as:

---

#### Algorithm Genetic Algorithm

Steps for Genetic Algorithm for rank aggregation (Kaur et al., 2017)

---

**Start** : Set GA parameters and initial random population

- 1: Execute algorithm and invoke objective functions such as Kendall-tau (KTu) and Spearman's footrule as (SFD) (Beg and Ahmad 2003)
- 2: Iteration: set  $k=1$ 
  - (a) select, crossover and mutate // GA step (Goldberg, 1989)
  - (b) compute parameter for distance measures
  - (c) compute  $f_{\min}(\text{distance})$

If  $(k+1) < n$  or  $f_{\min}(\text{distance}) > \text{tolerance}$ ; Goto 3
- 3: Execute GA with a new set of parameters as per 3 and compute performance measures.

**end.**

---

Equations (5) and (6) give the absolute difference between the ranks of  $k$  obtained according to the two lists. The two given lists are  $\sigma$ ,  $\tau$ , and their Spearman's footrule distance is given as:

$$F(\sigma, \tau) = \sum_k |\sigma(k) - \tau(k)| \quad (6)$$

The objective function from Eq. (7) is evaluated with the selected set of chromosomes from the pool of randomly generated permutations of the universe  $U = \{1, 2, 3, \dots, |U|\}$ . Then, each of the chromosomes from the pool takes its turn to evaluate the objective function, which is the aggregated footrule distance and the aggregated Kendall distance of that chromosome with the rankings given by the  $M$  participating search engines. Those chromosomes having less values sur-

vive, whereas a more value-based chromosomes leaves the pool (Beg and Ahmad 2003).

During the crossover stage, we will consider two to-be-crossed chromosomes as  $D_i^k = [d_{(i,1)}, d_{(i,2)}, \dots, d_{(i,n)}]$  and  $D_j^k = [d_{(j,1)}, d_{(j,2)}, \dots, d_{(j,n)}]$  where  $n = |U|$ .

Then, exchange of elements within the group of first  $m$  ( $m < n$ ) elements will take place, and remaining elements which are  $d_{(i,m+1)}, d_{(i,m+2)}, \dots, d_{(i,n)}$  in  $D_i^k$  and  $d_{(j,m+1)}, d_{(j,m+2)}, \dots, d_{(j,n)}$  in  $D_j^k$  will get adjusted to form new sets of chromosomes which are  $[d_{j,1}, \dots, d_{j,m}, d'_{i,m+1}, \dots, d'_{i,n}]$  and  $D_j^{(k+1)} = [d_{(i,1)}, \dots, d_{(i,m)}, d'_{(j,m+1)(j,m+1)}, \dots, d'_{(j,n)}]$  to form valid permutations. The implementation of permutation will take by using the concept of multiplication of permutations (Beg and Ahmad 2003). The following chromosomes will be formed after multiplication of permutation as per the relations formulated as.

$$\{d'_{i,m+1}, \dots, d'_{i,n}\} = \{d_{(j,m+1)}, d_{(i,m+2)}, \dots, d_{(j,n)}\} \\ \times \{d_{(i,m+1)}, d_{(i,m+2)}, \dots, d_{(i,n)}\} \quad (8)$$

$$\{d'_{j,m+1}, \dots, d'_{j,n}\} = \{d_{(i,m+1)}, d_{(i,m+2)}, \dots, d_{(j,n)}\} \\ \times \{d_{(j,m+1)}, d_{(j,m+2)}, \dots, d_{(j,n)}\} \quad (9)$$

New generation-based valid permutation of the universe  $U$  for two distinct chromosomes will be formed as  $D_i^{k+1}$  and  $D_j^{k+1}$ . The digit to-be-mutated is exchanged with any other randomly selected digit in that very permutation. This process will result in producing chromosomes of valid permutation (Beg and Ahmad 2003).

GA (Kaur et al. 2017) was applied for finding the optimal values of Kendall-tau (KTu) and Spearman's footrule (SFD) distance measures. Also, the initial population is created with help of rank aggregation methods such as Borda, MBV, Markov, PageRank and scaled footrule methods given in Table 2. Then, the Kendall-tau and Spearman's footrule are chosen as objective functions in order to calculate the distance value. The results obtained from 50 and 100 iterations using GA is shown in Tables 3 and 4. The complete steps for the proposed GA algorithm using Kendall-tau and Spearman's footrule for rank aggregation are discussed in Kaur et al. (2017).

From the literature, a lot of drawbacks such as large execution time, the choice of rate of mutation, crossover and selection are need to be carefully handled for GA to avoid false results. The performance of individual query is also not covered in our previous work (Kaur et al. 2017) in terms of Precision, Recall and  $F$ -Measure. Here we proposed ant colony-based Kendall-tau and Spearman's footrule distance measures for rank aggregation to each query. The performance of individual search engine pertaining to each query with and without word association is also tested using the Precision, Recall and  $F$ -Measure parameters.

## 5 Proposed ant colony-based rank aggregation strategy

In 1992, M. Dorigo had discovered the foraging behavior of wild ants by noting their food searching capabilities. Later on this foraging behavior was utilized for solving various optimization problems reported by Dorigo (1992) and Dorigo and Gambardella (1997). Ant colony optimization strategy is mainly inspired by the food searching technique of ants by strengthening the pheromone concentration on the path closest to the food and evaporation of pheromone of pheromones from the path far away from the food/destination Yang (2014). ACO was also, applied for solving discrete problems addressed by Dorigo et al. (1999) and Blum (2005). Similarly, the natural behavior of honey bees for food search is given by Gould and Gould (1988). Primarily, the ACO is based on the principle that at first, a random route is generated which is similar to mutation. Then pheromone based selection helps in selecting the shortest routes. In ACO there is no explicit crossover but fitness based mutation helps in providing the solutions. In this work a web link rank aggregation problem is framed and two objective functions are defined such as Kendall tau and spearman foot rule. The URLs have been arranged in a manner that URL with minimum objective function values will occupy highest or first rank position in the whole list among other URLs. The pseudo code for proposed Ant Colony based rank aggregation strategy is given as:



**Algorithm** Ant Colony Optimization

Steps for proposed ACO based rank aggregation strategy

*Start*

**Step 1.** Initialize the parameters: Select the random number of an artificial ant operators (Initial positions of URLs are given by the search engines for a particular query which are further refined by rank aggregation methods to new rank positions as indicated by the digits in Table 2). Set the iteration count as  $k = T$ .

**Step 2.** Calculate the value of objective function as given by Eq.(7)

**Step 3.** All the rank positions of URLs for each query are examined with respect to the reference aggregated list for  $T$  number of iterations. For  $k = k + 1$  the rank positions of URLs are moved to new positions.

**Step 4.** The probability of movement of artificial ants from one position to other so as to minimize the objective function (minima of Kendall Tau and Spearman's foot rule distance measure) is followed by the relation [Dorigo et al. \(1999\)](#) in terms of URLs position pool, pheromone coefficient and distance between the positions as:

$$P_s(l, p) = \begin{cases} \frac{h_{lp}^\gamma \times \gamma_{lp}^\eta}{\sum_{p \in D_s(l)} h_{lr}^\gamma \times \gamma_{pr}^\eta} & \\ 0 & \end{cases} \quad (10)$$

Where:

- $P_s(l, p)$  is the probability of movement by the ant  $s$  from position  $l$  to position  $p$ .
- $p \in D_s(l)$  Includes the set of positions which are yet to be visited by ant  $s$ .
- $\gamma$  the pheromone trail coefficient.
- $\eta$  is the coefficient of distance between positions.

**Step 5.** The criterion for selecting new rank position of a URL is based on the probability of movement of an ant which is determined by the objective function as per Eq.(7), [Kaur et al. \(2017\)](#) which indicates the position and concentration of pheromone present in the trial.

**Step 6.** In the pool of URLs positions when the ant  $s$  has visited all the positions and the corresponding distance values are obtained with the help of objective function, a new aggregated rank list of URLs has been derived on the basis of minimum distance values. Depending upon the pheromone concentration deposition, all the URLs are being arranged in such a way that minimum the distance value assigned with higher the rank position to the URL is followed by the relation [Dorigo et al. \(1999\)](#) given as:

$$\sigma(l, p) = P \cdot \sigma(l, p) + \sum_{q=1}^r \sigma_q(l, p) \quad (11)$$

$$\sigma_q = \begin{cases} \frac{1}{Z_q}, (l, p) \in \text{total positions} & \\ 0, \text{if otherwise} & \end{cases} \quad (12)$$

where:

- $P \cdot \sigma(l, p)$  is the density of pheromones lies between the  $l$  and  $p$  positions of ants which is multiplied with evaporation constant  $\xi$  and value lies between 0 and 1. If the concentration of pheromone is low then that distance value is discarded.
- $\sum_{q=1}^r \Delta \sigma_q(l, p)$  is the pheromone intensity deposited on the edge by an ant  $q$ , which is expressed by path length of ant visits to all positions and denoted by  $Z_q$ . So, on the basis of minimum objective function values (because short distance values have more pheromone density) all the URLs are arranged to provide an optimized rank list.

**Step 7.** If iteration count  $k + 1 = T$   
*end*  
 If  $k + 1 \neq T$   
 Go to step 2.  
*End*

**Table 1** Main queries and new logical operator-based Queries

Query no.	Query
Q1.	Parallel sorting neural networks
Q1a.	Parallel sorting <b>and</b> neural networks
Q1b.	Parallel sorting <b>or</b> neural networks
Q2.	Parallel architecture computing
Q2a.	Parallel architecture <b>and</b> computing
Q2b.	Parallel architecture <b>or</b> computing
Q3.	Computer information technology
Q3a.	Computer <b>and</b> information technology
Q3b.	Computer <b>or</b> information technology

The optimized values of objective functions obtained with the help of proposed ACO based strategy are depicted in the Tables 3 and 4 respectively.

## 6 Results and discussion

In this section, the performance of proposed ant colony-based strategy is compared with genetic algorithm for rank aggregation. In this work, initial dataset is taken and referred from Kaur et al. (2017) which comprised of three main queries (Q1–Q3) which are further refined by using word association as shown in Table 1. All these queries are fed to the five search engines such as *Google*<sup>1</sup>, *AltaVista*<sup>2</sup>, *Deeperweb*<sup>3</sup>, *Excite*<sup>4</sup> and *Hotbot*<sup>5</sup>.

The aggregated rank list is given in Table 2 for each query by rank aggregation methods such as Borda, mean-by-variance, Markov Chain MC4, PageRank and scaled footrule.

The comparative analysis between GA- and ACO-based ranking strategies is discussed in the next subsections.

### 6.1 Performance comparison of GA- and ACO-based ranking strategies

Both the proposed ACO- and GA (Kaur et al. 2017)-based ranking strategies are compared on the basis of experimental simulation results for 50 and 100 number of iterations. Table 3 indicates the values of optimized values of Kendall-tau distance measure and execution time at 100 iterations for each query by rank aggregation methods. The values of 50 iterations are not indicated due to space constraints. But the graphical results for 50 iterations are shown in Figs. 1 and 2, respectively.

Likewise, the optimized values of Spearman's footrule distance measure and execution time for both GA- and proposed ACO-based techniques are shown in Table 4 at 100 iterations.

Tables 3 and 4 show the optimized distance values for Kendall-tau and Spearman's footrule distance measure with

standard form, GA and ACO at 100 iterations, respectively. In Table 3, it is observed that proposed ACO-based strategy gives better results as compared to GA. For example, for Query 1 the Kendall-tau distance value is 0.237 with standard method, with GA it is 0.226, and by using ACO it is reduced to 0.178 with very less execution time as 13.16 s. Also, at 100th iteration Spearman's footrule value is 0.51 with GA, whereas with ACO the value is 0.31 with execution time of 14.98 s compared to 139.8 s with GA as shown in Table 4. Hence, from the overall analysis, the optimized results by proposed ACO-based strategy are found much better than GA and standard form for both the distance measure.

However, average aggregated distance value for Borda method with standard form is found minimum, but ACO-based strategy gives optimized results for other ranking methods. Table 5 shows the average aggregated rank of all queries for Kendall-tau and Spearman's distance measures.

Table 6 indicates the average aggregated rank values of Kendall Tau and Spearman Foot Rule for Genetic and proposed ACO based ranking strategies. The aggregated distance values for all the ranking methods have been found minimum with proposed strategy as compared to Genetic optimization (Kaur et al. 2017) as indicated in Table 6. It is concluded from the aggregated ranks that the Borda method has minimum rank value among its counterparts. Hence, the Borda method has been chosen for further analysis and for obtaining much refined rank list which can meet the user's satisfaction level.

Figure 1 indicates the graphical representation of best cost derived from GA and proposed ACO approach for Borda method with Query Q1 at 50th and 100th iterations. Figure 2 signifies that ACO-based approach outcast GA in terms of best cost, i.e., the optimized distance measure value. Also, Fig. 2 represents the best cost for Borda method with Query Q3b at 100 iterations which gets improved from 50th iteration with the proposed ACO algorithm.

### 6.2 Best optimized aggregated rank list for Queries

For the given set of three queries Q1-Q3, to test the relevancy of a query two sub queries have been derived using AND-OR technique as indicated in Table 1, which consist of a total of nine queries. Hence, from the above analysis for selection of best query among the set of three queries like from the set of queries Q1, Q1a and Q1b, the query Q1 has occupied the top most position in the aggregated list with minimum distance value by Borda's method as indicated in Tables 3 and 4. Table 7 shows the previous and optimized new ranks of 20 URLs for query Q1. Hence, the final aggregated list of 20 URLs for query Q1 is

Q1:{1, 3, 2, 4, 7, 9, 16, 19, 12, 18, 5, 10, 6, 11, 15, 8, 13, 14, 17, 20} as given in Table 7.

**Table 2** Aggregated rank list for each query by rank aggregation methods

Rank aggregation method	Aggregated list
Borda method	$t_{\text{query1}} = \{1, 3, 2, 4, 7, 9, 16, 19, 12, 20, 5, 10, 6, 11, 15, 8, 13, 14, 17, 20\}$ $t_{\text{query1a}} = \{3, 1, 4, 2, 16, 5, 17, 6, 12, 13, 8, 11, 7, 9, 15, 10, 14, 18, 19, 20\}$ $t_{\text{query1b}} = \{1, 3, 5, 4, 2, 18, 7, 16, 14, 19, 12, 6, 13, 11, 8, 9, 10, 15, 17, 20\}$ $t_{\text{query2}} = \{1, 2, 8, 4, 6, 9, 7, 10, 5, 13, 3, 11, 12, 14, 15, 16, 17, 18, 19, 20\}$ $t_{\text{query2a}} = \{1, 2, 9, 3, 5, 6, 7, 15, 10, 4, 8, 13, 11, 12, 14, 16, 17, 18, 19, 20\}$ $t_{\text{query2b}} = \{2, 1, 3, 4, 5, 8, 6, 9, 10, 13, 7, 1, 12, 15, 14, 16, 17, 18, 19, 20\}$ $t_{\text{query3}} = \{11, 5, 4, 2, 12, 7, 8, 10, 6, 9, 1, 3, 19, 14, 15, 16, 13, 17, 18, 20\}$ $t_{\text{query3a}} = \{8, 5, 1, 2, 3, 14, 10, 6, 7, 16, 9, 12, 11, 4, 20, 15, 13, 17, 18, 19\}$ $t_{\text{query3b}} = \{2, 1, 16, 3, 6, 4, 12, 5, 7, 10, 11, 8, 9, 13, 14, 18, 15, 20, 17, 19\}$
Mean-by-variance	$t_{\text{query1}} = \{19, 20, 16, 10, 11, 9, 15, 7, 4, 3, 12, 5, 6, 1, 2, 8, 13, 14, 17, 18\}$ $t_{\text{query1a}} = \{17, 15, 9, 13, 2, 6, 8, 1, 12, 11, 5, 16, 4, 10, 3, 7, 14, 18, 19, 20\}$ $t_{\text{query1b}} = \{19, 16, 18, 11, 13, 14, 2, 12, 6, 7, 8, 9, 10, 5, 3, 4, 1, 15, 17, 20\}$ $t_{\text{query2}} = \{13, 11, 12, 4, 7, 6, 9, 10, 1, 2, 8, 5, 3, 15, 14, 16, 17, 18, 19, 20\}$ $t_{\text{query2a}} = \{11, 15, 8, 13, 7, 6, 5, 9, 3, 12, 10, 4, 14, 1, 2, 16, 17, 18, 19, 20\}$ $t_{\text{query2b}} = \{13, 4, 15, 9, 7, 10, 1, 6, 5, 12, 14, 3, 8, 11, 2, 16, 17, 18, 19, 20\}$ $t_{\text{query3}} = \{4, 16, 2, 11, 10, 19, 12, 9, 15, 1, 3, 14, 6, 7, 8, 13, 5, 17, 18, 20\}$ $t_{\text{query3a}} = \{1, 4, 16, 20, 15, 12, 14, 2, 6, 17, 8, 3, 7, 11, 9, 13, 10, 19, 5, 18\}$ $t_{\text{query3b}} = \{16, 20, 7, 11, 10, 12, 8, 5, 13, 4, 18, 3, 9, 15, 17, 1, 6, 14, 19, 2\}$
Markov chain MC4	$t_{\text{query1}} = \{1, 18, 2, 3, 17, 14, 13, 19, 4, 7, 16, 11, 20, 8, 9, 15, 5, 6, 12, 10\}$ $t_{\text{query1a}} = \{1, 3, 20, 2, 19, 4, 18, 15, 17, 5, 14, 10, 16, 9, 12, 13, 6, 7, 8, 11\}$ $t_{\text{query1b}} = \{1, 3, 15, 17, 20, 5, 4, 8, 10, 11, 2, 9, 16, 19, 6, 7, 8, 14, 11, 13\}$ $t_{\text{query2}} = \{2, 1, 20, 7, 19, 18, 6, 8, 17, 4, 9, 16, 15, 5, 10, 12, 14, 11, 13, 3\}$ $t_{\text{query2a}} = \{1, 2, 20, 3, 5, 19, 9, 17, 18, 6, 16, 14, 15, 7, 12, 4, 10, 11, 8, 13\}$ $t_{\text{query2b}} = \{1, 2, 20, 3, 19, 4, 18, 5, 17, 6, 16, 8, 15, 14, 7, 9, 13, 10, 12, 11\}$ $t_{\text{query3}} = \{4, 20, 5, 11, 2, 18, 12, 17, 19, 7, 16, 8, 15, 14, 6, 3, 13, 1, 9, 10\}$ $t_{\text{query3a}} = \{8, 1, 5, 19, 18, 17, 2, 3, 13, 14, 10, 15, 20, 6, 16, 9, 12, 4, 7, 11\}$ $t_{\text{query3b}} = \{2, 1, 20, 16, 19, 18, 6, 17, 3, 12, 15, 4, 14, 5, 7, 13, 11, 8, 9, 10\}$
PageRank method	$t_{\text{query1}} = \{18, 17, 14, 13, 11, 15, 12, 8, 10, 6, 5, 20, 9, 16, 7, 19, 4, 2, 3, 1\}$ $t_{\text{query1a}} = \{20, 19, 18, 15, 14, 10, 9, 7, 8, 11, 6, 13, 12, 16, 5, 17, 4, 3, 2, 1\}$ $t_{\text{query1b}} = \{15, 17, 20, 16, 10, 11, 9, 8, 6, 14, 13, 7, 12, 19, 2, 18, 4, 5, 3, 1\}$ $t_{\text{query2}} = \{20, 19, 18, 17, 16, 15, 14, 12, 11, 3, 13, 10, 5, 9, 4, 8, 2, 6, 7, 1\}$ $t_{\text{query2a}} = \{20, 19, 18, 17, 16, 4, 1, 12, 11, 13, 8, 4, 10, 7, 15, 2, 6, 9, 3, 5, 1\}$ $t_{\text{query2b}} = \{20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 7, 8, 2, 6, 5, 4, 3, 1\}$ $t_{\text{query3}} = \{20, 19, 18, 17, 16, 1, 3, 15, 14, 9, 13, 10, 6, 8, 7, 12, 2, 11, 5, 4\}$ $t_{\text{query3a}} = \{18, 20, 17, 16, 19, 4, 15, 14, 12, 11, 7, 9, 6, 10, 13, 3, 2, 1, 5, 8\}$ $t_{\text{query3b}} = \{20, 19, 18, 17, 15, 14, 13, 11, 10, 9, 8, 7, 5, 12, 4, 3, 6, 1, 2, 16\}$
Scaled footrule (SFO)	$t_{\text{query1}} = \{19, 20, 16, 6, 11, 5, 12, 10, 15, 4, 2, 7, 9, 3, 13, 14, 3, 17, 18, 1\}$ $t_{\text{query1a}} = \{17, 16, 7, 9, 10, 12, 13, 2, 8, 6, 15, 5, 1, 11, 14, 4, 3, 18, 19, 20\}$ $t_{\text{query1b}} = \{18, 19, 16, 6, 8, 9, 10, 14, 2, 11, 13, 12, 7, 15, 5, 17, 3, 4, 1, 20\}$ $t_{\text{query2}} = \{3, 13, 7, 4, 9, 5, 10, 12, 2, 8, 11, 6, 1, 15, 14, 16, 17, 18, 19, 20\}$ $t_{\text{query2a}} = \{15, 4, 9, 13, 8, 7, 6, 11, 3, 10, 5, 12, 14, 1, 2, 16, 17, 18, 19, 20\}$ $t_{\text{query2b}} = \{13, 7, 4, 9, 6, 10, 1, 3, 5, 8, 12, 14, 11, 15, 2, 16, 17, 18, 19, 20\}$ $t_{\text{query3}} = \{1, 3, 11, 19, 12, 4, 10, 2, 6, 9, 14, 15, 16, 13, 8, 5, 7, 17, 18, 20\}$ $t_{\text{query3a}} = \{4, 14, 20, 8, 16, 2, 1, 3, 10, 6, 7, 9, 12, 11, 15, 13, 5, 17, 18, 19\}$ $t_{\text{query3b}} = \{16, 12, 7, 5, 4, 11, 3, 8, 20, 10, 9, 15, 13, 18, 1, 17, 2, 14, 6, 19\}$

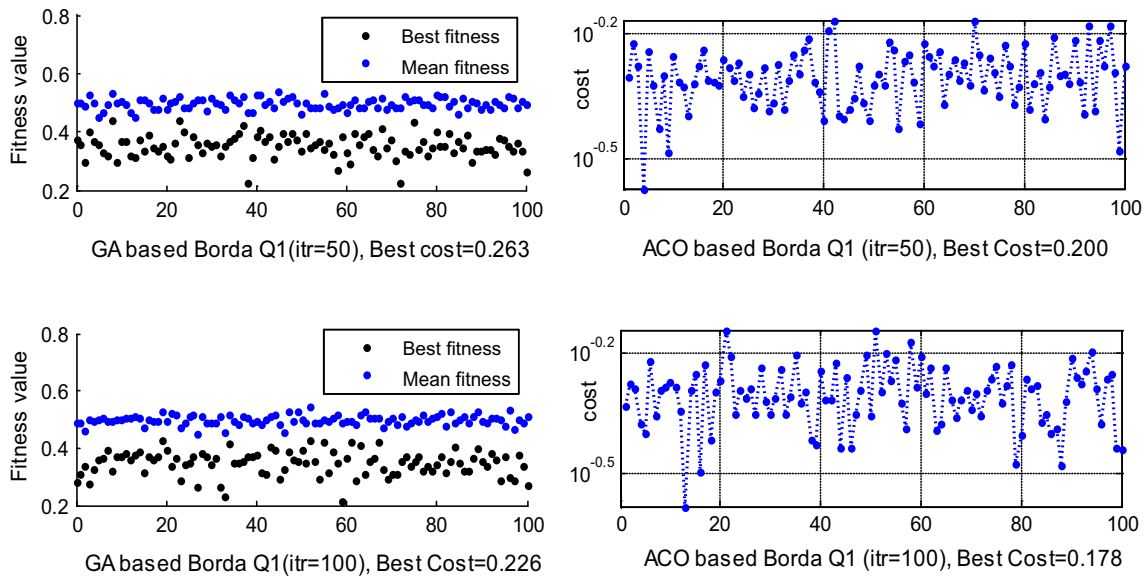


**Table 3** Comparison of proposed ACO and GA optimized Kendall-tau distance measure at (100 gen.)

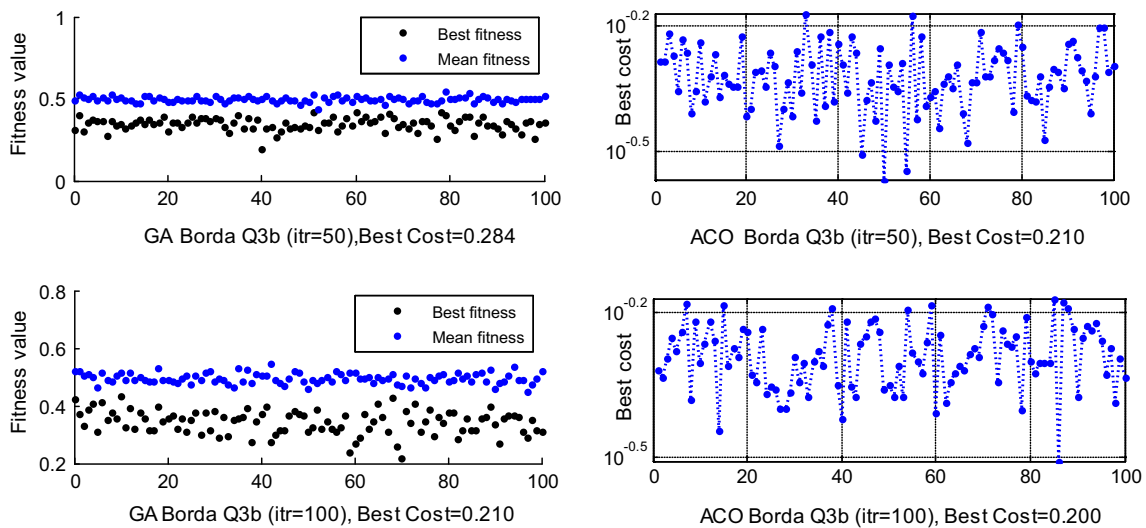
Query no.	Kendall-tau distance obtained after aggregating the ranks using				
	Standard	Soft computing algorithms with 100 generation			
		GA	Execution time (s)	ACO	Execution time (s)
<b>Borda method</b>					
Q1	0.237	0.226	259.0	0.178	13.617
Q1a	0.211	0.211	264.6	0.247	13.053
Q1b	0.284	0.280	300.0	0.236	13.069
Q2	0.100	0.305	147.0	0.226	15.731
Q2a	0.105	0.370	273.0	0.221	13.923
Q2b	0.047	0.380	276.0	0.232	15.556
Q3	0.253	0.389	207.0	0.205	15.184
Q3a	0.215	0.384	228.0	0.189	14.837
Q3b	0.158	0.240	252.0	0.200	15.088
<b>MBV method</b>					
Q1	0.568	0.320	228.0	0.189	14.445
Q1a	0.421	0.320	228.0	0.232	14.259
Q1b	0.611	0.305	258.0	0.221	14.456
Q2	0.305	0.331	267.0	0.205	14.573
Q2a	0.378	0.345	258.8	0.226	14.674
Q2b	0.321	0.332	207.0	0.205	14.662
Q3	0.400	0.342	139.8	0.211	14.802
Q3a	0.447	0.331	336.0	0.200	15.285
Q3b	0.563	0.341	324.0	0.232	14.877
<b>Markov chain method</b>					
Q1	0.474	0.300	324.0	0.290	15.672
Q1a	0.511	0.367	207.0	0.300	15.608
Q1b	0.400	0.420	336.2	0.280	15.857
Q2	0.500	0.233	270.0	0.300	15.716
Q2a	0.468	0.268	288.0	0.330	15.256
Q2b	0.436	0.350	320.0	0.310	15.749
Q3	0.574	0.380	312.0	0.290	15.885
Q3a	0.495	0.370	312.0	0.300	15.831
Q3b	0.553	0.280	270.0	0.250	15.647
<b>PageRank method</b>					
Q1	0.758	0.320	207.0	0.280	16.344
Q1a	0.795	0.330	273.2	0.331	16.008
Q1b	0.732	0.300	160.0	0.292	15.507
Q2	0.895	0.310	194.0	0.311	15.838
Q2a	0.853	0.310	153.0	0.430	15.345
Q2b	0.974	0.240	270.0	0.301	22.560
Q3	0.774	0.330	300.0	0.332	15.765
Q3a	0.822	0.350	273.0	0.310	15.632
Q3b	0.868	0.310	313.2	0.301	16.043
<b>Scaled footrule method</b>					
Q1	0.600	0.336	267.0	0.261	15.322
Q1a	0.478	0.370	207.0	0.290	16.230
Q1b	0.600	0.330	300.6	0.322	15.634

**Table 3** continued

Query no.	Kendall-tau distance obtained after aggregating the ranks using				
	Standard	Soft computing algorithms with 100 generation			
		GA	Execution time (s)	ACO	Execution time (s)
Q2	0.226	0.360	147.8	0.260	14.908
Q2a	0.342	0.360	168.8	0.311	15.513
Q2b	0.236	0.340	207.6	0.321	15.789
Q3	0.316	0.400	199.8	0.282	15.668
Q3a	0.336	0.310	147.0	0.272	15.266
Q3b	0.453	0.300	153.0	0.241	15.263



**Fig. 1** Graphical representation of fitness value at 50 and 100 iterations for GA and ACO to query Q1 by Borda



**Fig. 2** Graphical representation of fitness value at 50 and 100 iterations for GA and ACO to query Q3b by Borda

**Table 4** Comparison of proposed ACO and GA optimized Spearman's footrule distance measures at (100 gen.)

Query no.	Spearman's footrule distance obtained after aggregating the ranks				
	Standard	Soft computing algorithms with 100 generation			
		GA	Execution time (s)	ACO	Execution time (s)
<b>Borda method</b>					
Q1	0.380	0.510	139.8	0.310	14.98
Q1a	0.300	0.520	192.0	0.330	16.25
Q1b	0.380	0.530	180.0	0.300	15.67
Q2	0.140	0.480	144.0	0.310	15.45
Q2a	0.150	0.530	258.0	0.320	15.63
Q2b	0.090	0.520	253.8	0.290	15.96
Q3	0.310	0.570	240.0	0.270	16.25
Q3a	0.320	0.530	293.8	0.310	15.56
Q3b	0.240	0.520	220.2	0.310	16.25
<b>MBV method</b>					
Q1	0.730	0.420	206.4	0.270	15.789
Q1a	0.530	0.400	270.0	0.320	15.497
Q1b	0.740	0.400	216.0	0.300	15.653
Q2	0.370	0.500	325.8	0.270	15.949
Q2a	0.430	0.480	330.0	0.250	15.583
Q2b	0.420	0.580	216.0	0.320	16.314
Q3	0.560	0.430	280.0	0.280	15.378
Q3a	0.620	0.520	264.0	0.310	15.794
Q3b	0.700	0.410	210.0	0.330	16.140
<b>Markov chain method</b>					
Q1	0.650	0.460	336.0	0.291	15.672
Q1a	0.640	0.450	273.0	0.300	15.608
Q1b	0.565	0.380	276.0	0.280	15.857
Q2	0.610	0.390	207.0	0.300	15.716
Q2a	0.590	0.520	206.5	0.330	15.256
Q2b	0.570	0.440	153.6	0.310	15.749
Q3	0.730	0.500	300.0	0.290	15.885
Q3a	0.670	0.470	207.0	0.300	15.831
Q3b	0.720	0.560	253.8	0.250	15.647
<b>PageRank method</b>					
Q1	0.860	0.300	240.0	0.280	16.344
Q1a	0.820	0.380	226.9	0.330	16.008
Q1b	0.830	0.400	228.0	0.290	15.507
Q2	0.990	0.400	260.4	0.310	15.838
Q2a	0.960	0.535	288.0	0.430	15.345
Q2b	1.000	0.440	193.8	0.300	22.560
Q3	0.890	0.470	276.0	0.330	15.765
Q3a	0.930	0.460	270.0	0.310	15.632
Q3b	0.920	0.440	300.0	0.300	16.043

**Table 4** continued

Query no.	Spearman's footrule distance obtained after aggregating the ranks				
	Standard	Soft computing algorithms with 100 generation			
		GA	Execution time (s)	ACO	Execution time (s)
Scaled footrule method					
Q1	0.725	0.435	204.0	0.261	15.322
Q1a	0.600	0.450	213.0	0.290	16.230
Q1b	0.690	0.480	270.0	0.321	15.634
Q2	0.290	0.520	260.0	0.260	14.908
Q2a	0.390	0.450	273.0	0.310	15.513
Q2b	0.320	0.520	216.0	0.320	15.789
Q3	0.530	0.520	288.0	0.280	15.668
Q3a	0.480	0.420	234.0	0.271	15.266
Q3b	0.580	0.550	193.8	0.240	15.263

**Table 5** Performance comparison of rank aggregation methods on the basis of aggregated ranks

Aggregation techniques	Avg. aggregated distance [0, 1] Kendall-tau (all queries)	Avg. aggregated distance [0, 1] Spearman's footrule (all queries)
Borda method	0.268	0.385
Mean-by-variance (MBV)	0.669	0.850
Markov MC4	0.735	0.957
PageRank	1.245	1.306
Scaled footrule (SFO)	0.597	0.765

**Table 6** Comparison of rank aggregation methods on the basis of GA and ACO optimized Kendall and Spearman's distance measures

Rank aggregation methods	Kendall-tau	Spearman's footrule	Kendall-tau	Spearman's footrule
	GA strategy (Kaur et al. 2017)	GA strategy (Kaur et al. 2017)	Proposed ACO rank strategy	Proposed ACO rank strategy
Borda method	0.411	0.785	0.322	0.458
Mean-by-variance (MBV)	0.495	0.690	0.321	0.442
Markov MC4	0.494	0.695	0.442	0.443
PageRank	0.467	0.638	0.48	0.48
Scaled footrule (SFO)	0.517	0.724	0.425	0.425

Similarly, among other queries such as Q2, Q2a and Q2b, the query Q2a is having more optimized aggregated list of URLs from scaled footrule rank aggregation method as given in Tables 3 and 4. Out of queries Q3, Q3a and Q3b, the query Q3a of PageRank is the winner on the basis of minimum distance measure value obtained from proposed ACO-based strategy given in Tables 3 and 4. Hence, the best final aggregated URL list for Q2a and Q3a is:

$$t_{\text{query2a}} = \{15, 4, 9, 13, 8, 7, 6, 11, 3, 10, 5, 12, 14, 1, 2, 16, 17, 18, 19, 20\}$$

$$t_{\text{query3a}} = \{4, 14, 20, 8, 16, 2, 1, 3, 10, 6, 7, 9, 12, 11, 15, 13, 5, 17, 18, 19\}$$

### 6.3 Performance metrics for optimized lists

In this section, the various testing parameters such as Precision, Recall and  $F$ -Measures are applied to investigate the performance of metasearch engines.

#### 6.3.1 Precision metrics

The Precision metrics is also called as positive predictive value which is the fraction of retrieved instances that are relevant. The performance of final aggregated queries such as query Q1, Q2a and Q3a is evaluated on the basis of precision. Precision is the ratio of retrieved relevant document

**Table 7** Optimized aggregated rank list of URLs

Old rank	Optimized aggregated list-based URLs of Q1 for Borda	New rank
1	<a href="http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=62417">http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=62417</a>	1
3	<a href="http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.13.2304&amp;rep=rep1&amp;type=pdf">http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.13.2304&amp;rep=rep1&amp;type=pdf</a>	2
2	<a href="http://link.springer.com/article/10.1007%2FBF00228719">http://link.springer.com/article/10.1007%2FBF00228719</a>	3
4	<a href="http://core.ac.uk/download/pdf/13372685.pdf">http://core.ac.uk/download/pdf/13372685.pdf</a>	4
7	<a href="http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol1/cs11/article1.html">http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol1/cs11/article1.html</a>	5
9	<a href="http://dl.acm.org/citation.cfm?id=2689060">http://dl.acm.org/citation.cfm?id=2689060</a>	6
16	<a href="http://www.personal.psu.edu/lnl/papers/aiaa20057168.pdf">http://www.personal.psu.edu/lnl/papers/aiaa20057168.pdf</a>	7
19	<a href="https://en.wikipedia.org/wiki/Artificial_neural_network">https://en.wikipedia.org/wiki/Artificial_neural_network</a>	8
12	<a href="http://in.mathworks.com/help/nnet/ug/neural-networks-with-parallel-and-gpu-computing.html">http://in.mathworks.com/help/nnet/ug/neural-networks-with-parallel-and-gpu-computing.html</a>	9
18	<a href="http://www.neurones.espci.fr/denby/ieee_tnn_14_5_2003.pdf">http://www.neurones.espci.fr/denby/ieee_tnn_14_5_2003.pdf</a>	10
5	<a href="http://dosen.narotama.ac.id/wp-content/uploads/2013/02/7.-Parallelization-of-Algorithms-with-Recurrent-Neural-Networks.pdf">http://dosen.narotama.ac.id/wp-content/uploads/2013/02/7.-Parallelization-of-Algorithms-with-Recurrent-Neural-Networks.pdf</a>	11
10	<a href="http://motherboard.vice.com/read/why-googles-neural-networks-look-like-theyre-on-acid">http://motherboard.vice.com/read/why-googles-neural-networks-look-like-theyre-on-acid</a>	12
6	<a href="http://research.microsoft.com/pubs/175561/asru-2011.pdf">http://research.microsoft.com/pubs/175561/asru-2011.pdf</a>	13
11	<a href="http://www.cs.tau.ac.il/~wolf/deeplearningmeeting/pdfs/Sup+LSTM.pdf">http://www.cs.tau.ac.il/~wolf/deeplearningmeeting/pdfs/Sup+LSTM.pdf</a>	14
15	<a href="http://www.cs.stir.ac.uk/~lss/NNIntro/InvSlides.html">http://www.cs.stir.ac.uk/~lss/NNIntro/InvSlides.html</a>	15
8	<a href="http://www.gcn.us.es/6BWMC/volume/sn-snps-7.pdf">http://www.gcn.us.es/6BWMC/volume/sn-snps-7.pdf</a>	16
13	<a href="http://digitalcommons.calpoly.edu/">http://digitalcommons.calpoly.edu/</a>	17
14	<a href="http://www.eecs.berkeley.edu/~krste/papers/Mstrain.pdf">http://www.eecs.berkeley.edu/~krste/papers/Mstrain.pdf</a>	18
17	<a href="http://www.d.umn.edu/~tkwon/research/papers/KWTA_networks.pdf">http://www.d.umn.edu/~tkwon/research/papers/KWTA_networks.pdf</a>	19
20	<a href="http://www.neuro.sfc.keio.ac.jp/publications/pdf/hys.pdf">http://www.neuro.sfc.keio.ac.jp/publications/pdf/hys.pdf</a>	20

over retrieved documents. The Precision value for the URLs derived for query Q1 is calculated using Eq. (13). Table 8 represents the precision for top twenty documents retrieved from all the five search engines.

$$\text{Precision} = \frac{\text{retrieved relevant}}{\text{retrieved}} = \frac{|A \cap B|}{|B|} \quad (13)$$

In Table 8, the average precision for Google and Excite search engines is found best than for other search engines.

### 6.3.2 Recall metrics

Recall is the ratio of retrieving relevant documents over the relevant documents. The Recall metrics is evaluated for Q1 which is the fraction of relevant instances that are retrieved also known as sensitivity. Using Eq. (14), the value of Recall is calculated for Q1 and results are shown in Table 9. The value of recall for first URL of Q1 is 0.05 which means that out of total 20 URLs (relevant documents retrieved), the first URL is 0.05 times relevant, i.e., (1/20). Similarly, for all top twenty documents the Recall is calculated as shown in Table 9.

$$\text{Recall} = \frac{\text{retrieved relevant}}{\text{relevant}} = \frac{|A \cap B|}{|A|} \quad (14)$$

### 6.3.3 F-Measure metrics

F-Measure is also known as F-score which is the weighted harmonic mean of measuring the accuracy. The F-Measure is used during the work where Precision and Recall are equally weighted. It considers both the value of Precision and Recall to calculate the score. F-Measure is calculated using Eq. (15) for top twenty documents retrieved for Query Q1 as shown in Table 10.

$$F\text{-Measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (15)$$

Table 10 represents the F-Measure for Query 1 calculated using Eq. (15) for twenty URLs.

Figure 3 shows the comparison of Precision, Recall and F-Measure on the basis of average value of performance metrics for the query Q1. Also, Fig. 3 represents that out of all five search engines, Google performance metrics out-cast the performance of other search engines. The positions



**Table 8** Precision for Query Q1

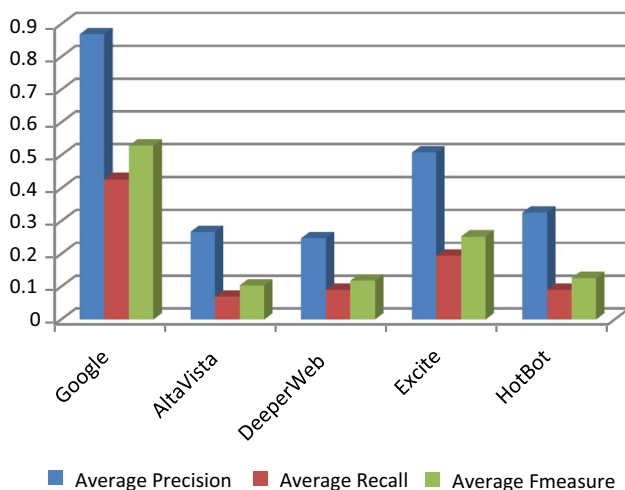
Precision (p)_Q1	Google	AltaVista	DeeperWeb	Excite	HotBot
p_1	1.00	1.00	1.00	1.00	1.00
p_5	1.00	0.60	0.60	1.00	1.00
p_10	0.00	0.70	0.00	0.80	0.00
p_15	0.93	0.00	0.47	0.67	0.00
p_20	0.00	0.00	0.00	0.00	0.00
Average	0.586	0.460	0.414	0.694	0.400

**Table 9** Recall for Query Q1

Recall (r)_Q1	Google	AltaVista	DeeperWeb	Excite	HotBot
r_1	0.05	0.05	0.05	0.05	0.05
r_5	0.25	0.15	0.15	0.25	0.00
r_10	0.00	0.35	0.00	0.40	0.00
r_15	0.70	0.00	0.35	0.50	0.00
r_20	0.00	0.00	0.00	0.00	0.00
Average	0.20	0.22	0.11	0.24	0.01

**Table 10** *F*-Measure for Query Q1

<i>F</i> -Measure ( <i>f</i> )_Q1	Google	AltaVista	DeeperWeb	Excite	HotBot
f_1	0.05	0.05	0.05	0.05	0.05
f_5	0.25	0.15	0.15	0.25	0.00
f_10	0.00	0.35	0.00	0.40	0.00
f_15	0.70	0.00	0.35	0.50	0.00
f_20	0.00	0.00	0.00	0.00	0.00
Average	0.20	0.22	0.11	0.24	0.01

**Fig. 3** Average performance metrics

acquired by search engines in terms of performance metrics are Google > Excite > HotBot > AltaVista > Deeperweb.

The values for the Precision, Recall and *F*-Measure for all the final aggregated queries in order to check the relevancy of the documents are shown in Table 11. Also, Table 11 repre-

sents that the average performance metrics for Precision and Recall which are used to check the relevancy of each query as given in Tables 3 and 4 by the proposed ACO scheme. Hence, it is concluded from the performance analysis that Google and Excite search engines are the winners in producing most relevant documents to the users by the proposed ACO-based strategy.

## 7 Conclusion and future scope

In this work, ranking problem of web links has been framed by applying different rank aggregation techniques. Kendall tau and Spearman's footrule distance measures based objective functions are chosen for optimization purpose. Ant Colony based rank optimization strategy has been proposed to minimize the objective function value and to assign highest rank to a web link having lowest objective function value. Also, the performance of the proposed ranking strategy has been compared with the Standard and GA optimized rank aggregation methods on the basis of distance values (Kendall and Spearman) and execution time for each query as given in Tables 3 and 4 respectively. From the simulation results it has

**Table 11** Average performance metrics

	Google	AltaVista	HotBot	Excite	Deeper web	Google	AltaVista	HotBot	Excite	Deeper web
	Average Precision (Kendall-tau)					Average Recall (Kendall-tau)				
Q1	0.87	0.27	0.33	0.51	0.25	0.43	0.07	0.09	0.195	0.09
Q2a	0.34	0.16	0.24	0.28	0.34	0.14	0.07	0.05	0.09	0.13
Q3a	0.66	0.19	0.27	0.22	0.40	0.28	0.05	0.14	0.05	0.07
	Average Precision (Spearman's footrule)					Average Recall (Spearman's footrule)				
Q1	0.74	0.30	0.26	0.29	0.30	0.34	0.09	0.07	0.09	0.13
Q2a	0.61	0.38	0.21	0.25	0.52	0.3	0.17	0.037	0.07	0.19
Q3a	0.49	0.09	0.22	0.02	0.49	0.17	0.04	0.04	0.02	0.19

been concluded that Borda is the best method among its other aggregation methods as it gives minimum distance values. At last performance tests are applied such as Precision, Recall and *F*-Measure to test the proposed strategy. Also, the word association operators are applied to the queries which provide more prominent and very close results to the web lists provided by search engines. In future, other meta-heuristic approaches such as Particle Swarm, Teacher-learner, Moth search and hybrid techniques can be applied to optimize the performance of search engines and to improve metasearch techniques.

**Acknowledgements** The contributions of all the authors are highly obliged for their continuous work and efforts. Also, the contributions of cited authors are highly appreciated for their research findings and quality of work. The work has been conducted at I.K.G Punjab Technical University, Jalandhar.

#### Compliance with ethical standards

**Ethical approval** This article does not contain any studies with human participants or animals performed by any of the authors.

**Conflict of interest** Authors declare that they have no conflict of interest.

## References

- Arrow K (1951) Social choice and individual values. Wiley, New York
- Akritidis L, Katsaros D, Bozaris P (2011) Effective rank aggregation for metasearching. *J Syst Softw* 84:130–143
- Beg MMS, Ahmad N (2002) Fuzzy logic based rank aggregation methods for the world wide web. In: Proceedings of the international conference on artificial intelligence in engineering and technology, Malaysia, pp 363–368
- Beg MMS, Ahmad N (2003) Soft computing techniques for rank aggregation on the world wide web. *J Int Inf Syst* 6(1):5–22
- Beg MMS, Ahmad N (2004) Study of rank aggregation for World Wide Web. *J Study Fuzziness Soft Comput* 137:24–46
- Beg MMS, Ansari MZ, Kumar M (2016) Enhancement of fuzzy rank aggregation technique. Proceedings of the second international conference on computer and communication technologies, advances in intelligent systems and computing 381:127–135
- Blum C (2005) Ant colony optimization: introduction and recent trends. *J Phys Life Rev* 2:353–373
- Borda JC (1781) Mémoire sur les élections au scrutin. Histoire de l'Académie Royale des Sciences
- Diaconis P, Graham R (1977) Spearman's footrule as a measure of disarray. *J R Stat Soc B* 39(2):262–268
- Diaconis P (1988) Group representation in probability and statistics. IMS lecture series 11, Institute of Mathematical Statistics, Hayward
- Dorigo M (1992) Optimization, learning and natural algorithms. Ph.D. thesis, Politecnico di Milano
- Dorigo M, Gambardella LM (1997) Ant colony system: a cooperative learning approach to the travelling salesman problem. *Proc IEEE Trans Evolut Comput* 1(1):53–66
- Dorigo M, Di Caro G, Gambardella LM (1999) Ant algorithms for discrete optimization. *Artif Life* 5(2):137–172
- Dwork C, Kumar R, Naor M, Sivakumar D (2001) Rank aggregation methods for the Web. In: Proceedings of tenth ACM international conference on World Wide Web, pp 613–622
- Goldberg DE (1989) Book on genetic algorithms in search, optimization and machine learning. Addison Wesley, Reading
- Gould JL, Gould CG (1988) The honey bee. Scientific American Library, New York
- Holland J (1975) Adaptation in natural and artificial systems. University of Michigan Press, Ann Arbor
- Kaur M, Kaur P, Singh M (2015) Rank aggregation using multi objective genetic algorithm. In: Proceedings of IEEE sponsored international conference on next generation technologies, Dehradun, pp 836–840
- Kaur P, Singh M, Josan GS (2017) Comparative analysis of Rank Aggregation techniques for metasearch using genetic algorithm. *J Educ Inf Technol* 22(3):965–983
- Kemeny JL, Snell JG (1962) Mathematical models in the social sciences. Ph.D. thesis, New York
- Kendall MG (1938) A new measure of rank correlation. *Biometrika* 30(1/2):81–93
- Kleinberg J (1999) Authoritative sources in a hyperlinked environment. *ACM* 46(5):604–632
- Langville AN, Meyer CD (2012) Who's #1?: The science of rating and ranking. Princeton University Press, Princeton
- Laughlin A, Olson J, Simpson D, Inoue A (2011) Page ranking refinement using fuzzy sets and logic. In: Proceedings of the 22nd midwest artificial intelligence and cognitive science conference, Cincinnati, pp 40–46
- Montague MH, Aslam JA (2001) Models of metasearch. In: Proceedings of the ACM international conference on research and development in information retrieval (SIGIR), pp 276–284
- Napoles G, Dikopoulou Z, Papegeorgiou E, Bello R, Vanhoof K (2015) Aggregation of partial rankings—an approach based on the Kemeny

- ranking problem. *Adv Comput Intell Lect Notes Comput Sci* 9095:343–355
- Page L, Brin L (1998) The anatomy of a large-scale hyper textual web search engine. In: *Proceedings of seventh international World Wide Web conference*
- Pihur V, Datta S, Datta S (2014) RankAggreg, an R package for weighted rank aggregation. A report by Department of Bioinformatics and Biostatistics, University of Louisville. <http://vpihur.com/biostat>
- Renda ME, Straccia U (2003) *Web metasearch: rank vs. score based rank aggregation methods*. SAC, Melbourne
- Ross TJ (1997) *Book on fuzzy logic with engineering applications*. McGraw-Hill, New York
- Saari DJ (2000) The mathematics of voting: democratic symmetry. *Economist*.83. <http://vidal.ece.sc.edu/822/papers/econ-voting.html>
- Shimura M (1973) Fuzzy sets concept in rank-ordering objects. *J Math Anal Appl* 43:717–733
- Waad B, Atef BB, Mohamed L (2013) Feature selection by rank aggregation and genetic algorithms. In: *Proceedings of the international conference on knowledge discovery and information retrieval, Vilamoura*, pp 74–81
- Yang XS (2010) *Engineering optimization: an introduction with meta-heuristic applications*. Wiley, London
- Yang XS (2014) *Nature inspired optimization algorithms*, 1st edn. Elsevier, Amsterdam
- Yan L, Gui C, Du W, Guo Q (2011) An improved PageRank method based on genetic algorithm for web search. In: *Proceedings of advanced in control engineering and information science procedia engineering* 15, pp 2983–2987. Elsevier