8

Ritu Meena\* and Kamal K. Bharadwaj\*

# A Genetic Algorithm Approach for Group Recommender System Based on Partial Rankings

https://doi.org/10.1515/jisys-2017-0561 Received October 10, 2017; previously published online June 20, 2018.

**Abstract:** Many recommender systems frequently make suggestions for group consumable items to the individual users. There has been much work done in group recommender systems (GRSs) with full ranking, but partial ranking (PR) where items are partially ranked still remains a challenge. The ultimate objective of this work is to propose rank aggregation technique for effectively handling the PR problem. Additionally, in real applications, most of the studies have focused on PR without ties (PRWOT). However, the rankings may have ties where some items are placed in the same position, but where some items are partially ranked to be aggregated may not be permutations. In this work, in order to handle problem of PR in GRS for PRWOT and PR with ties (PRWT), we propose a novel approach to GRS based on genetic algorithm (GA) where for PRWOT Spearman foot rule distance and for PRWT Kendall tau distance with bucket order are used as fitness functions. Experimental results are presented that clearly demonstrate that our proposed GRS based on GA for PRWOT (GRS-GA-PRWOT) and PRWT (GRS-GA-PRWOT) outperforms well-known baseline GRS techniques.

**Keywords:** Group recommender systems, partial ranking, rank aggregation, genetic algorithm, bucket order.

# 1 Introduction

Recommender systems (RSs) are one of the information overloaded web personalization tools to manage the web and provide users those items that best fit their individual needs. The most successful and widely used filtering technique is collaborative filtering. Most recommender systems make recommendations of the group consumable selected items (e.g. movies) to the individual. Recently there have been several efforts in the area of group recommender systems (GRSs).

GRSs [4, 15] consider preferences of each member of a group and provide such recommendations to groups so that suggested list of items (or a single recommended item) satisfies the group members optimally. However, the optimal solution is not possible so that most of the GRS proposed are providing near-optimal solutions [2, 5]. The main concern is how to effectively aggregate the preferences of individuals in a group to produce recommendations that will satisfy a group of users.

The goal of rank aggregation (RA) is to find out an aggregate ranking so that the distance to each of the given partial ranked list get minimized. In the domain of GRS ranking can be categorized as follows:

- Full rankings: Item list is fully ranked by a member of the group.
- Partial rankings (PRs): Item list is partially ranked, i.e. some items in the list are unranked by members of the group.

RA is a useful abstraction with a number of applications which includes meta-search, similarity search algorithms, composite rank functions from multiple listing, and classification. For GRSs, RA techniques, such as Most pleasure, Least misery, Borda count, and Copeland rule have been successfully used [9, 10, 12].

<sup>\*</sup>Corresponding authors: Ritu Meena and Kamal K. Bharadwaj, School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi, India, e-mail: meena.ritu@gmail.com (R. Meena); kbharadwaj@gmail.com (K.K. Bharadwaj)

In real applications, the rankings which are partial (all items are not ranked by users) and are to be aggregated may not be permutations, but they may have ties where some elements are placed at the same position. However, PR with ties is more challenging as we cannot mathematically compare two PRs, which are rankings that have ties in the rank list [8, 11].

There are a number of literature available in the domain of GRS with full rankings, but PR is relatively much less explored [10, 11]. Among various aggregation strategies, Spearman foot rule aggregation is supposed to be a unique procedure that generates best compromise ordering that minimizes the sum of Spearman foot rule distance (SFD) from the input orderings. In general, this minimization problem is (non-deterministic polynomial-time hardness) NP-hard, and therefore a genetic algorithm (GA) is most suited for producing a near-optimal solution [18, 20].

In this paper, both the PR without ties (PRWOT) and PR with ties (PRWT) are considered to develop novel GRS Schemes, GRS-GA-PRWOT and GRS-GA-PRWT, employing GA. In our GA approach, fitness functions SFD for PRWOT and Kendall tau distance (KtD) with bucket order for PRWT were used. Experimental results are demonstrating the effectiveness of the proposed GRS-GA-PRWOT and GRS-GA-PRWT Schemes.

Section 2 of this article presents related work, and the proposed GRS-based approaches using GA for PRWOT (GRS-GA-PRWOT) and PRWT (GRS-GA-PRWT) are discussed in Section 3. Details of the experiments and results are established in Section 4. Finally, in Section 5 we conclude the paper and outline some direction for future work.

### 2 Related Works

RSs are a method to solve the problem of information overload and successfully employ a variety of choices by combining ideas from user preferences, filtering information, and providing the user more intelligent and proactive information by making concrete item recommendations.

In this section GRSs, RA, PRWOT, GA, SFD measure, KtD using bucket order for PRWT, and effectiveness of rank list of recommendation are discussed [4, 9].

#### 2.1 Group Recommender Systems

GRSs make suggestions for a group. It considers the preferences of each individual member of a group into account and provides an optimal single recommendation. "In order to generate effective recommendations for a group, the system must satisfy, as much as possible, the individual preferences of the group's members" [9, 21].

GRSs approaches can be classified into two broad categories [1, 4]. The first one merges the profiles of multiple users and constructs a single preference model for the group, and the other one makes the group recommendation instead of the individual's recommendations. The RA is a prominent method for developing multiple strategies for GRS [5, 19].

#### 2.2 Rank Aggregation

There are two well-established ranking methods, namely, KtD and SFD. Fagin et al. has proposed various metrics for comparing PR-suggested algorithms for computing few topmost items of a near-optimal aggregation of multiple different PRs [8, 10–12].

#### 2.3 Partial Ranking without Ties

Following Dwork et al. [10] and Fagin et al. [12], we encounter PRs where there are some items unranked by a set of users, and such lists that rank only some of the items by a set of users are called *partial list*. There are many situations where a full listis not convenient or even cannot be possible. A special case of partial lists is the following. If  $\sigma$  is a set of rank order for a full list which includes all results, another set is called  $\tau$  which is

a partial list which is some subset of  $\sigma$ ,  $\tau$  is only a subset of  $\sigma$ , and each ranked item list above all unranked items is called top *n* list, where *n* is the size of the list. In order to deal with PRs, modified SFD is used [8, 11].

### 2.4 Partial Ranking with Ties

Let us consider the problem of RA of PRWT, that is, where items have the same preference for a user can be tied. Following Brancotte et al. [8], a bucket order on n is represented by a set of buckets  $B_1$  to  $B_m$  that forms a disjointed partition of *n* such that *x* has transitive binary relation with *y* if there are *i*, *j* with i < j such that  $x \in B_i$  and  $y \in B_i$ . A ranking with ties on n is defined as  $r = [B_1, \dots, B_m]$ , where r[x] = i if  $x \in B_i$ . Although the classical formulation of the KtD allows comparing rankings with ties, in this case, it is not a distance anymore. Moreover, ties are actually ignored, and no disagreement can be considered for items which do not have ties. Thus, independent of the input, the ranking with the fewest disagreements is the ranking where all elements are tied in a unique bucket. To avoid reducing such a useless solution, algorithms based on KtD have to restrain themselves to produce permutations [11, 12].

With reference to only one ranking, a set of items which are either counter or tied would be counted as one disagreement. KtD is equivalent to sorting the elements and can be done, with adaptations, when considering ranking with ties [8, 11]. An optimal consensus ranking of a set of rankings with ties  $R \subseteq Rn$  under the generalized Kemeny RA with ties is a ranking such that

$$\forall \in \text{Rn: } K(r, R < K(r, R)).$$

A consensus ranking denotes a not certainly optimal solution of the problem. When a solution is optimal, it is explicitly denoted as an optimal solution of a problem [6, 14].

### 2.5 Bucket Order (Spearman Foot Rule Distance Measure)

A bucket order is a linear order intuitively, with ties where every bucket is of size 1. Since the foot rule optimal aggregation is an optimal distance measure, the technique is called Kemeny optimal aggregation. The computations of optimal aggregation using foot rule for the partial list is an NP-hard problem of computing distance between one partial list and full list example of bucket order show in Figure 1.

#### 2.6 Genetic Algorithm

A population of chromosomes is processed in a GA approach where a possible solution to an optimization problem is shown by chromosomes, and the solution quality is measured using a fitness function. Chromosomes with minimum distance fitness value are selected, and offspring are produced using genetic operators, crossover, and mutation [19, 20]. A new population is formed by replacing individuals with low fitness in the current populations by the newly generated good individuals.

That will be acceptable stopping criterion, such as for termination of GA when a maximum number of generations proceed or a desired level of fitness is reached will be a suitable stopping criterion. GA is the suitable technique to produce near-optimal solution for Kemeny optimal aggregation [7, 18]. Main steps of GA are summarized below:

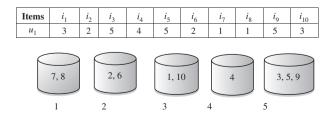


Figure 1: Bucket Order Representation.

- (i) Create a population of random individuals in which each individual represents a possible solution to the problem at hand.
- (ii) Evaluate each individual's fitness its ability to solve the specified problem.
- (iii) Select individual population members to be parents.
- (iv) Produce offspring by recombining parent material via crossover and mutation and add them to the population.
- (v) Evaluate the offspring's fitness.
- (vi) Repeat steps iii–v until a solution with the desired fitness goal is obtained.

#### 2.7 The Effectiveness of a Ranked List of Recommendation

Normalized discounted cumulative gain (nDCG) is used to evaluate the effectiveness of a ranked list of recommendations [4]. Computation of nDCG requires full ratings, and updating of nDCG for partial ratings is utilized in our work.

For example, if  $I = \{1, 3, 5, 9, 2, 7, 6, 4, 8\}$  is a recommendations list of rank for group and user u test set is {1, 4, 9, 2, 3}, then nDCG is computed on the ranked list {1, 3, 9, 2, 4}.

# 3 The Proposed PR-Based Approach for GRSs

A GRSs approach based on GA for PR is presented in this work.

#### **Data Encoding**

Let us suppose that there is a group of m users and n items. Example of partial ranking matrices is given in Table 1

Group = 
$$\{u_1, u_2, \dots, u_m\}$$
  
Items =  $\{i_1, i_2, \dots, i_n\}$ 

#### **3.1 Scheme 1**

Assuming that a group of *n* users do not give their preferences for all *m* items, not all items are present in the list; the rating matrix  $m^*n$  is generated in which the *i*th row represents the rating of  $u_i$  for  $i_1$ , to  $i_n$ .

Here, to solve the problem of GRS, generate a group of 15 items for aggregating m chromosomes that tries to satisfy different sets of users optimally.

In view of the limited access of GRS data sets, we have done a test to conduct our experiments on a randomly generated data set that closely relates the characteristics of the information available on the internet sources like MovieLens and GoupLens data sets. Schematic representation of our proposed Scheme 1 is show in Figure 2.

Details of the proposed GRS-GA-PRWOT based on PRWOT are given below:

#### 3.1.1 Metrics for PRWOT

Metrics on PRs without ties are defined below.

This is the matrix we have to compare with full ranking using fitness formula that is SFD.

Table 1: Example of Partial Ranking Matrices.

Items	1	2	3	4	5	6	7	8	9	10
$\overline{u_1}$	-	8	6	-	-	9	5	-	1	4
$u_1$	7	6	10	2	9	_	4	8	3	_

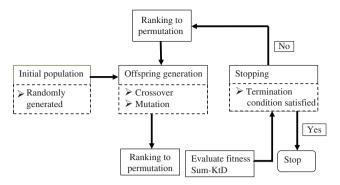


Figure 2: The Schematic Representation of Our Proposed GRS-GA-PRWOT.

#### 3.1.2 Fitness Function for PRWOT (Sum-SFD)

Given a partial list  $\tau$  and a full list  $\sigma$ , the SFD is explained in Table 2 (Example of sum-SFD)

$$G(\sigma, \tau) = \sum_{i \in \tau} |\sigma(i)/\sigma - \tau(i)/|\tau||$$

$$\sigma = 4, 1, 2, 9, 7, 8, 3, 5, 6, 10$$
(1)

Our proposed GRS finally recommends the list of items that is the permutation of best chromosome permutation with the minimum SFD.

#### **Encoding**

A chromosome for a group of users is simply the permutation of items, i.e.  $(i_8, i_7, \dots, i_3)$  which 

Offspring Generation. The following genetic sequencing operators are employed to generate offspring [18, 20].

### **Uniform Order-Based Crossover. Explained in Figure 4** Scramble Sublist Mutation show in Figure 5

Selection Criteria. In order to avoid the possibility of crossover or mutation destroying best individuals, elitist approach is used, where top chromosomes are preserved from one generation to the next generation.

Table 2: Example of Fitness Function (Sum-SFD).

Ranking											SFD												
R1	5	-	3	-	-	6	7	-	10	-	3.9												
R2	3	_	4	8	_	7	2	_	_	9	1.9												
R3	-	8	_	10	-	7	6	-	1	2	4.2												
R4	5	_	10	_	_	8	_	3	_	1	4.1												
R5	-	-	7	-	8	5	-	2	9	-	3.6												
	Total	distance (	sum-SFD)							Total distance (sum-SFD)													

Items	1	2	3	4	5	6	7	8	9	10
$u_1$	$i_8$	$i_7$	$i_5$	$i_9$	$i_{10}$	$i_2$	$i_1$	$i_6$	$i_5$	$i_3$
$u_1$	$i_{_4}$	$i_9$	$i_5$	$i_1$	$i_6$	i <sub>10</sub>	$i_8$	$i_2$	$i_3$	$i_7$

Is represent on

Items	1	2	3	4	5	6	7	8	9	10
$u_1$	8	7	5	9	10	2	1	6	5	3
$u_1$	4	9	5	1	6	10	8	2	3	7

Figure 3: Representation of Chromosomes.

Each bit is selected from one of corresponding genes of parent chromosomes.

	Parent 1	4	5	2	1	7	3	6	9	10	8
	Parent 2	1	2	3	4	5	6	7	8	9	10
Γ	Bit string	1	0	0	1	1	0	1	0	1	1

Wherever the bit string contains a "1" some of the positions on Offspring 1 is filling by copying them from Parent 1 and fill the positions on Offspring 2 wherever the bit string contains a "0".

	Offspring 1	4			1	7		6		10	8
ſ	Offspring 2		2	3			6		8		

Make a list of items from Parent 1 associated with a "0" in the bit string. [5, 2, 3, 9] are ordered according to Parent 2 [2, 3, 5, 9]



To make Offspring 2 carry out a similar process. [1, 4, 5, 7, 9, 10] are ordered according to Parent 1 [4, 5, 1, 7, 9, 10]



Figure 4: Uniform Order-Based Crossover.

Select individual and chose a sublist randomly

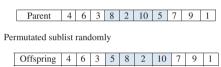


Figure 5: Show Scramble Sublist Mutation.

#### 3.2 Scheme 2

Here, for solving the problem of GRS is to generate a group of 15 items for aggregating m chromosome that tries to satisfy different sets of users optimally. Movie lens datasets used for Scheme 2.

Details of the proposed GRS-GA-PRWT based on partial rating with ties are given below. Schematic representation of our proposed Scheme 2 is show in Figure 6.

#### 3.2.1 Metrics for PRWT

**Bucket Order.** For each bucket of size 1, a bucket order is termed as a linear order [12].

Let  $\mathcal{B}_1 \dots \mathcal{B}_m$  be the bucket in order that bucket  $\mathcal{B}_i$  precedes bucket  $\mathcal{B}_j$  when i < j example of bucket order show in Figure 1.

This is the matrix we have to compare with full ranking with ties ( $\sigma$ ). These are located on a concept of the KtD to top k list.

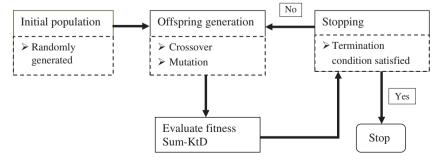


Figure 6: The Schematic Representation of Our Proposed GRS-GA-PRWT.

Here,  $\sigma$  is full ranking with ties, and  $\tau$  is PRWT.  $\mathcal{P}$  is a penalty, with  $0 \leq \mathcal{P} \leq 1$ . According to the definition a penalty score  $\overline{k}_{i,j}^{(p)}(\sigma,\tau)$  is defined for PRs  $\tau_1,\tau_2.....\tau_n$ ,  $\{i,j\}\in\mathcal{P}$ . We calculate the penalty using three cases.

**Case 1:** If *i* and *j* are in different order (bucket) in both  $\tau$  and  $\sigma$  or if *i* and *j* are in the same order in  $\sigma$  and  $\tau$ (such as  $\tau(i) > \tau(j)$  and  $\tau(i) > \sigma(j)$ ), and both rankings agree that i and j are tied, then

$$\overline{k}_{i,j}^{(p)}(\sigma,\tau)=0;$$

**Case 2:** If *i* and *j* are in the opposite bucket in  $\sigma$  and  $\tau$  (such as  $\sigma(i) > \sigma(j)$  and  $\tau(i) < \tau(j)$ ), then

$$\overline{k}_{i,j}^{(p)}(\sigma,\tau)=1.$$

**Case 3:** If *i* and *j* are in the same order in one of the rankings  $\sigma$  and  $\tau$ , but in a different order in the other PR, then in this case,

$$\overline{k}_{i,j}^{(p)}(\sigma,\tau)=p.$$

#### 3.2.2 Fitness Function for PRWT (Sum-KtD)

Based on these three cases, define k(p), the KtD with penalty parameter p, sum-KTD with bucket order explained in Table 3 as follows:

$$k^{(p)}(\sigma,\tau) = \sum_{\{i,j\}\in\mathcal{P}} \overline{k}_{i,j}^{(p)}(\sigma,\tau).$$

$$\sigma = 2, 4, 3, 1, 5, 4, 3, 5, 4, 1$$
(2)

Crossover and Mutation

Here, we have to apply crossover and mutation in full ranking with ties.

#### **Single-Point Crossover**

One crossover point is selected randomly; the ranking is copied till this point from the first parent, then the second parent is scanned, and in the offspring it is added. Explained in Figure 7.

**Order-Changing Mutation.** In this mutation operator two numbers are selected randomly and exchanged with randomly generated numbers between 1 and 10. Explained in Figure 8.

Stopping Criteria for Genetic Algorithm. If there is no improvement in the fitness value after 20 consecutive generations, then evolution process terminates [16].

The Effectiveness of a Ranked List of Recommendations

Table 3: Example of Fitness Function (Sum-KtD with Bucket Order).

Ranking											KtD			
R1	5	-	3	-	-	4	2	-	5	-	3.5			
R2	3	-	4	2	_	5	2	_	-	1	4.5			
R3	_	5	-	2	_	1	3	_	1	2	7			
R4	5	-	1	-	_	4	_	3	-	1	6.5			
R5	_	_	2	-	3	5	_	2	4	_	5.5			
	Total	Total distance (sum-KtD)												

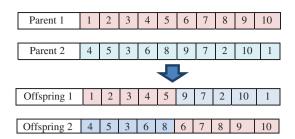


Figure 7: Example of Single-Point Crossover.

Parent	1	2	3	4	5	6	10	8	9	7
Offspring	1	8	3	4	5	6	10	2	9	7

Figure 8: Example of Order Changing Mutation.

Let *p*1, ..., *pk* be a recommended list of items produced by proposed GRSs model. Let *u* be a user and the actual rating is  $r_{upi}$  for the item pi of the user u ranked in position i, i.e.,  $\sigma_{u(pi)} = i$ . The definitions of nDCG, ideal discounted cumulative gain (IDCG) and discounted cumulative gain (DCG) at rank *k* are given below:

$$DCG_k^u = r_{up1} + \sum_{i=2}^k \frac{r_{upi}}{\log_2(i)}$$

$$nDCG_k^u = \frac{DCG_k^u}{IDCG_k^u}$$
(4)

$$nDCG_k^u = \frac{DCG_k^u}{IDCG_k^u} \tag{4}$$

where for user u the maximum possible gain value which is IDCG will be the same as DCG that is obtained with the optimal re-order of the k items in p1, ..., pk.

# 4 Experiments and Results

#### 4.1 Data Set

For our experiments, we have used both the synthetic data set and real-world MovieLens data set. We have performed experiments for Scheme 1 using synthetic data sets because the type of data required for this Scheme is not publicly available, whereas for Scheme 2 MovieLens data set is used, which contains 12,832 ratings provided by 1043 users for 1682 movies; we generated list of users who have rated at least 12 movies, which provide five random splits (S), S-1, S-2, S-3, S-4, and S-5, where for each split, 20 users were randomly selected as active users. The data sets consist of users' profiles and their preferences about movies.

#### **Scheme 1:** Partial Ranking without Ties (PRWOT).

In order to measure the performance of proposed GRS-GA-PRWOT Scheme with different baseline GRS techniques, our experiments had groups of different sizes (G5, G10, G15, and G20). The evaluation is based on the sum of Spearman foot rule distance (sum-SFD); the smaller the Sum-SFD, the better the proposed model.

Experiment 1. Here sum-SFD is computed for four different group sizes (G5, G10, G15, and G20) across generations. The results shown in Figure 9 clearly indicate that for all the groups of different sizes, GA converges to the near-optimal solution after 500 generations.

Experiment 2. In this experiment, the proposed GRS-GA-PRWOT is compared with the baseline techniques. The results depicted in Figure 10 clearly demonstrate that our Scheme GRS-GA-PRWOT outperforms several RA techniques given in the chart.

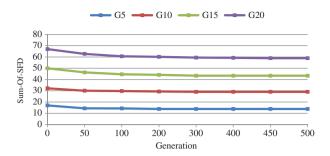


Figure 9: For Different Group Size Variation of Sum-SFD across Generations.

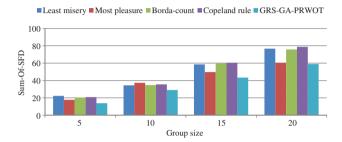


Figure 10: The Effectiveness of Proposed GRS-GA-PRWOT for Different Group Sizes Compared with Several Techniques.

Experiment 3. Here, the effectiveness of the group recommendation by our proposed Scheme GRS-GA-PRWOT is compared with baseline techniques based on mean nDCG with varying group sizes. Results are shown in Figure 11.

#### Scheme 2: Partial Ranking with Ties (PRWT).

In order to measure the performance of proposed GRS-GA-PRWT Scheme with different baseline GRS techniques, we conducted experiments with groups of different sizes (G10, G20, G30, G40, and G50) users and 20 items. The evaluation is based on the Kemeny optimal aggregation with bucket order (sum-KtD); the smaller sum-KtD shows that this technique outperforms several stats of art techniques.

Experiment 1. Here sum-SFD is computed for four different sizes of the group (G10, G20, G30, G40, and G50) across generations. The results shown in Figure 12 clearly indicate that for all the groups of different sizes, GA converges to the near-optimal solution after 500 generations.

Experiment 2. In this experiment, the proposed GRS-GA-PRWT is compared with the baseline techniques. The results depicted in Figure 13 clearly demonstrate that our Scheme GRS-GA-PRWT outperforms Least misery, Most pleasure, Borda count, and Copeland rule.

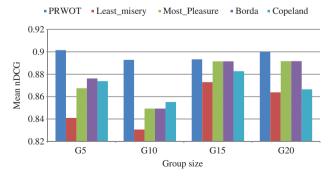


Figure 11: The Effectiveness of Group Recommendation with Rank Aggregation Techniques.

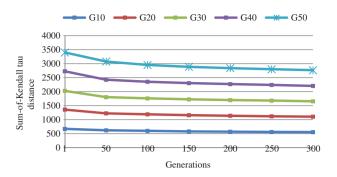


Figure 12: For Different Group Sizes Variation of Sum-KtD across Generations.

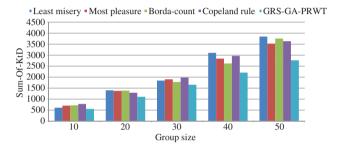


Figure 13: The Effectiveness of Proposed GRS-GA-PRWT Compared to Baseline Techniques Based on Different Group Sizes.

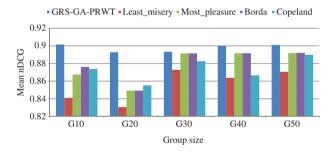


Figure 14: The Effectiveness of Proposed GRS-GA-PRWT Compared to Baseline Techniques Based on Mean nDCG for Different Group Sizes.

*Experiment 3.* Here, the effectiveness of the group recommendation by our proposed Scheme GRS-GA-PRWT is compared with baseline techniques based on mean nDCG with varying group sizes. Results are shown in Figure 14.

## 5 Conclusions and Future Work

A framework for GRSs is presented in this work where GA is employed successfully to deal with the aggregation problem of PRWOT as well as PRWT to generate near-optimal solutions, and two Schemes GRS-GA-PRWOT and GRS-GA-PRWT are proposed. Experimental results have clearly established the effectiveness of our proposed Schemes.

As a future research, we would like to investigate incorporation of negotiation mechanism [13, 14] and selected models and discuss their importance for recommender system development, notions of trust-distrust, and reputation [3, 17] into the proposed GRS techniques to further enhance their capability. Further, it remains to be seen how consensus-based GRS technique can improve the proposed Schemes [6, 14].

# **Bibliography**

- [1] G. Adomavicius and A. Tuzhilin, Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extension, IEEE Trans. Knowl. Data Eng. 17 (2005), 734-749.
- [2] D. Anand and K. K. Bharadwaj, Enhancing accuracy of recommender system through adaptive similarity measures based on hybrid features, ACIIDS 2 (2010), 1-10.
- [3] D. Anand and K. K. Bharadwaj, Pruning trust-distrust network via reliability and risk estimates for quality recommendations, Soc. Netw. Anal. Min. 3 (2013), 65-84.
- [4] L. Baltrunas, T. Makcinskas and F. Ricci, Group recommendations with rank aggregation and collaborative filtering, in: Proceedings of the 4th ACM Conference on Recommender Systems, pp. 119–126, 2010.
- [5] J. P. Baskin and S. Krishnamurthi, Preference aggregation in group recommender systems for committee decision-making, Rec Sys 09 (2009), 337-340.
- [6] D. Ben-Arieh and Z. Chen, Linguistic-labels aggregation and consensus measure for autocratic decision making using group recommendations, IEEE Trans. Syst. Man Cybern. A 36 (2006), 558-568.
- [7] K. K. Bharadwaj and M. Y. H. Al-Shamri, Fuzzy-genetic approach to recommender systems based on a novel hybrid user model, Expert Syst. Appl. 35 (2007), 1386-1399.
- [8] B. Brancotte, B. Yang, G. Blin, S. C. Boulakia, A. Denise and S. Hamel, Rank aggregation with ties: experiments and analysis, PVLDB 8 (2015), 1202-1213.
- [9] I. Cantador and P. Castells, Group recommender systems: new perspectives in the social web, in: Recommender Systems for the Social Web, Intelligent Systems Reference Library, vol 32. Springer, Berlin, Heidelberg, pp. 139-157, 2012.
- [10] C. Dwork, R. Kumar, M. Naor and D. Sivakumar, Rank aggregation methods for the web, in: Proceedings of the Tenth International Conference on the World Wide Web (WWW10), pp. 613-622, Hong Kong, 2001.
- [11] R. Fagin, R. Kumar, M. Mahdian, D. Sivakumar and E. Vee, Comparing and aggregating rankings with ties, PODS 2004 (2004), 47-58.
- [12] R. Fagin, R. Kumar, M. Mahdian, D. Sivakumar and E. Vee, Comparing partial rankings, SIAM J. Discrete Math. 20 (2006), 628-648.
- [13] I. Garcia, L. Sebastia and E. Onaindia, A negotiation approach for group recommendation, IC-AI 9662 (2009), 919-925.
- [14] I. Garcia, S. Pajares, L. Sebastia and E. Onaindia, Preference elicitation techniques for group recommender systems, Inf. Sci. 189 (2012), 155-175.
- [15] S. Herr, A. Rösch, C. Beckmann and T. Gross, Informing the design of group recommender systems, CHI Extended Abstracts (2012).
- [16] R. M. Jarvis and R. Goodacre, Genetic algorithm optimization for pre-processing and variable selection of spectroscopic data, Bioinformatics 21 (2005), 860-868.
- [17] V. Kant and K. K. Bharadwaj, Fuzzy computational models of trust and distrust for enhanced recommendations, Int. J. Intell. Syst. 28 (2013), 332-365.
- [18] D. Lawrence, Schedule optimization using genetic algorithms, in: V. Nostr and Reinhold, eds., Handbook of Genetic Algorithms, New York, 1991.
- [19] R. Meena and K. K. Bharadwaj, Group recommender system based on rank aggregation an evolutionary approach, in: Proceedings of the International Conference on Mining Intelligence and Knowledge Exploration (MIKE), LNCS 8284, Springer, pp. 663-676, 2013.
- [20] M. Mitchell, An Introduction to Genetic Algorithms. MIT Press, 1998, ISBN 978-0-262-63185-3, pp. I-VIII, 1-208.
- [21] M. O'Connor, D. Cosley, J. Konstan and J. Riedl, PolyLens: a recommender system for groups of users, in: Proceedings of the European Conference on Computer-Supported Cooperative Work, Mandl, 2001.