J. Intell. Syst. 2020; 29(1): 1092–1108 **DE GRUYTER**

Ritu Meena* and Sonajharia Minz

Group Recommender Systems – An Evolutionary Approach Based on Multi-expert System for Consensus

https://doi.org/10.1515/jisys-2018-0081
Received February 10, 2018; previously published online November 20, 2018.

Abstract: Recommender systems have focused on algorithms for a recommendation for individuals. However, in many domains, it may be recommending an item, for example, movies, restaurants etc. for a group of persons for which some remarkable group recommender systems (GRSs) has been developed. GRSs satisfy a group of people optimally by considering the equal weighting of the individual preferences. We have proposed a multi-expert scheme (MES) for group recommendation using genetic algorithm (GA) MES-GRS-GA that depends on consensus techniques to further improve group recommendations. In order to deal with this problem of GRS, we also propose a consensus scheme for GRSs where consensus from multiple experts are brought together to make a single recommended list of items in which each expert represents an individual inside the group. The proposed GA based consensus scheme is modeled as many consensus schemes within two phases. In the consensus phase, we have applied GA to obtain the maximum utility offer for each expert and generated the most appropriate rating for each item in the group. In the recommendation generation phase, again GA has been employed to produce the resulting group profile, i.e. the list of ratings with the minimum sum of distances from the group members. Finally, the results of computational experiments that bear close resemblance to real-world scenarios are presented and compared to baseline GRS techniques that illustrate the superiority of the proposed model.

Keywords: Group recommender systems, genetic algorithm, rank aggregation, consensus.

1 Introduction

In today's world more users become digitally linked so we have information surplus on the Internet. With the undue growth of information, it becomes very complex for a user to find the information he/she is seeking for [39, 40]. Internet search engines like Google and Bing are intended to provide useful and relevant information to a user, but they are also losing their significance because of the difficulty in finding useful information among thousands of results. This problem requires personalized web applications, which capture the most pertinent and valuable information from diverse sources. There is an enormous number of applications of web personalization in which recommender system (RS) [27, 36, 38] is the most notable application, which will make straightforward content detection and information access in a most valuable way that would be appropriate to users.

Generally, the aim of a traditional RS is to learn the behavior of users and, according to that, predict the rating of items and then suggest the items having a maximum rating that users might like [3, 7, 9]. But there are many such activities that involve a group of users, i.e. seeing a movie, going to a restaurant etc. [2, 6, 37]. I such a case, RSs that consider the taste and preferences of each member for the group of users are called group recommender systems (GRSs). This is an active research area in the field of RSs. Recently, a large number of GRSs have been developed in order to deal with the challenges of making recommendations for a group of members [10, 22].

***Corresponding author: Ritu Meena,** Jawaharlal Nehru University, New Delhi 110067, India, e-mail: meena.ritu@gmail.com
Sonajharia Minz: Jawaharlal Nehru University, New Delhi 110067, India

Some well-known GRSs are *Polylens, MESicFX, Intrigue, The Collaborative Advisory Travel System* and *Travel Decision Forum* [21] which recommend tourist attractions for a group of users and *GRSK* [17, 18].

The GRS is used to specify each user's preferences and then find out a compromise point on which all the members of the group agree equally. Most of the GRSs aggregate each member's preferences to generate the group profile; i.e. the system deals with each member's profile in the same way. This technique of GRSs has the following implications:

Mainly GRS techniques consider an equal weighting for all individual preferences while generating a group profile; i.e. all members are optimally satisfied. Every time the system deals with each member's preference in the same way, therefore, no member of the group is dissatisfied with the resulting group profile. However, these systems ignore the interaction of members with each other within the group [3, 11]. The new research in GRSs proves that it is more practical to consider each member's behavior and the connection between them.

Most of the GRSs use aggregation strategies to make a recommendation by considering each individual's preferences; i.e. each member has to communicate his/her preferences with the system. However, not all the users feel comfortable while sharing their preferences with the system, as this is the private information of every user in the group and it should not be made public. In other words, a GRS can also be treated as a distributed system, where only the system knows which information is shared by the user [1, 9, 13].

Briefly, a GRS multi-expert scheme (MES) considers the following issues:

– In a group recommendation problem, a single list of items is needed that fulfils the needs of all group members optimally.
– Every member of a group has different characteristics like individual preferences, relationship with other members in a group etc. Therefore, each member shows different behavior while making a recommendation as a whole.
– Every individual's preferences are private information, and the individual user makes a decision regarding what information to share with other members in the group.

Therefore, the success of a GRS is determined based on how individual preferences are aggregated into group preferences while considering the behavior of each member. To come up with the same preference, we have developed a *MES* for consensus [17, 28, 30]. The idea behind our proposed work is that each expert in a group maximizes his own utility (payoff) and reaches a mutually acceptable agreement. The consensus is a process of successive offers and counter-offers that takes place in a discrete number of consensus rounds. In this model, there are experts who act on behalf of each group member. Each expert in the group prepares the model of the respective user preferences, which includes the attributes and issues considered in consensus. Each expert communicates with all other experts in the group, and therefore this is not a centralized system [31, 37, 44].

The proposed *MES* model is implemented in two phases: consensus phase and recommendation generation phase. A genetic algorithm (GA) based MES for GRSs (GA-MES-GRS) is developed. This multilateral consensus scheme is modeled as many one-to-one bilateral consensus schemes. We have applied GA to obtain the maximum utility offer for each expert in the group [4, 5, 43, 47]. At the end of the consensus phase, we have generated the most appropriate rating for each user in the group. GA is also employed to produce a list of ratings with the minimum sum of distances in the recommendation generation phase. To summarize, our paper creates the following contributions:

– We propose a GA-MES-GRS scheme where multiple experts negotiate together to come up with a single recommended list of items.
– First of all, in consensus phase, we have used GA to find maximum utility offer for each expert in the group.
– Second, in recommendation generation phase, the GA has been also employed to find a list of ratings which minimizes the minimum sum of distances.
– Finally, we evaluate the effectiveness of GA-MES-GRS scheme for a group of users against baseline techniques of group recommendation.

The rest of the article is arranged as follows. The details of the related work are presented in Section 2. In Section 3 we have explained the proposed GA based approach to multi-expert based consensus process and depict the functionality of the application with an example. Section 4 gives details of various experiments conducted and the analysis of results obtained. In Section 5 we discuss the conclusions and outlined some future work in Section 6.

2 Related Work

In this section we discuss the previous research area related to our work; these can be written off as GRS techniques and consensus schemes for GRS and GA.

2.1 GRSs

Generally, the aim of a conventional RS is to study the activities of users and according to that predict the rating of items and then propose the items having maximum rating that users might like. The major task of GRS is to specify each user's preferences and then find out a compromise point on which all the members of the group agree equally. The idea behind group recommendation is to generate and aggregate the preferences of the individual user. As explained in [22] the major approaches to generating the preference aggregation are (a) merging the individual recommendations, (b) aggregation of individual rankings and (c) construction a group preference model. There are some GRSs that generate the group profile by considering equal weighting for all members in the group and avoid the interaction among them. Examples of such type of GRSs are *MusicFX*, *Let's Browse*, *Polylens*, *Intrigue*, *The Collaborative Advisory Travel System* and *GRSK* [2, 6, 14].

Furthermore, the group profile is not always generated by aggregating of all individual preferences as different members have different behavior. In recent years, several GRSs that consider each member's personality and social connection among them have been developed. For example, Quijano-Sanchez proposes a GRS that includes the personality and trust among group members in order to improve the quality of group recommendation. Furthermore, a memory of past recommendations is included to improve the accuracy of GRS that enhances the satisfaction of users whose preferences have not been considered in past recommendations. Another example of a GRS that focuses on the individual's personality is described in Masthoff [33] and *GRec-OC* [24, 31].

The baseline techniques of GRSs are Least Misery (LM), Most Pleasure (MP), Average Strategy, Borda Count etc. [7, 8, 15].

2.2 GA

GAs provide suitable resolution without requiring detailed structural information about the search space by mutually acceptable efficiency. Our proposed GA is used to drive the heuristic search methods to find near-optimal solutions for large search spaces in the context of consensus among the group, over a set of potential consensus solutions [25].

Evolutionary computation takes inspiration from the phenomenon of natural evolution and mimics the Darwinian idea of survival of the fittest. With their intrinsic capabilities, these evolutionary and heuristic strategies are popularly used to optimize problems, although the range of problems to which they are applied is quite wide [19, 25].

The selection operator chooses the chromosomes from the population on which genetic operators are applied. Various schemes are possible to choose the chromosomes as parents in the crossover such as roulette rank selection, tournament selection, Boltzmann selection and wheel selection. After selection, the crossover operator is applied on the chromosomes to get possibly better offspring. The aim of the crossover operator is to preserve and combine the best characteristics of the parents to evolve better new solutions. The crossover operator explores the search space in diverse directions to get different attractive solutions and allows genes from different parents to be combined to produce a single child. The competent potential of the offspring depends on the recombination operator used. Mutation operator is one of the most applied and widespread for introducing genetic diversity into the population at the level of the individual in the literature [25].

The next generation of n + 1 generation is generated by replacing individual preferences in the current population in generation n with the application of genetic operators. The empirical also uses an elite strategy where the best individuals of generation k are automatically transferred to the population of n + 1 generation, ensuring that the best solution found so far in the evolutionary process is always maintained by the algorithm [19, 32].

2.3 Consensus

Consensus refers to the process by which a group of experts communicate with one another to reach a mutually acceptable agreement [25].

A simple consensus space has 5- tuple $consensus = <P, A, D, U, T>$ where,

$P =$ finite set of consensus experts
$A =$ A set of attributes perceived by all the experts
$D =$ A set of attributes in the domain
$U =$ A set of utility functions, i.e. each function for an expert
$T =$ termination for each expert

The experts share information about a set of users, set of attributes and set of attribute domains before the consensus process is initiated. Numerous *bilateral consensuses* are used to model a *multilateral consensus* where in each turn an agent makes an offer for each of its opponents and stores the most beneficial counteroffer [23, 25, 29, 42].

The agreement is reached if one expert makes a proposal that is at least as good for each other expert as their own current proposal [35, 45, 49].

2.4 Consensus scheme for GRSs

This paper explains that a group of people is not able to use synchronous communication media. At any point of time, only one member can interact with the consensus system and the rest of the members of the group are represented by expert users. In the first phase, each member specifies its preferences, the evolution criteria of the utility of proposals and the relative utility of proposals for different members of the group. The aim of the second phase is to come up with a mutually acceptable agreement through consensus between the current user and the expert of absent users. Afterward,

(i) The experts compute an offer based on the specified choices of all members of the group.
(ii) The experts can accept or reject that offer according to the threshold specified by the equivalent real group member.
(iii) The current member replies to this offer by accepting or rejecting it or adapting his/her preferences, and the process goes back to step 1 again.

This process continues until the current member of the group agrees with the experts of other members or runs out of time or interest. In this system, the information about preferences of every individual is private to each one, and the decisions are made by real users.

The distributed structure of GRS facilitates the system to consider the behavior of different users. The GRS presented in [4, 20] depends on the application of cooperative consensus methodology to generate group recommendations. Here, each member of the group is characterized by an expert. Initially, the system Trip@dvice [39, 41] is used to generate individual recommendation lists, and later, the actual consensus process starts. Here, two consensus protocols are defined: Alternating Offers protocol, intended for direct consensus among two users, and Merging Ranks protocol, designed for consensus among more than two users.

There are several other recommendation techniques that use MES to generate a recommendation for a single user. For example, Garcia and Sebastia [16] describe a recommendation process for heterogeneous group members. A GA based adaptive consensus expert model is described in [12, 26, 45, 49].

3 Proposed Framework

This section provides details about the proposed GA based MES consensus scheme for GRSs (MES-GRS-GA). Basically, the aim of the consensus user is to maximize his own utility (payoff) and reach an agreement as soon as possible. The consensus users are sharing a common interest but have different user preferences. So, how they coordinate, how much the users reach consensus on some issue to maximize their own goal and come up with the mutually acceptable agreement is a great problem.

3.1 Representation of Proposed Model

This section provides details about the proposed multi-attribute consensus scheme based on a GA where multiple users work together to come up with a mutually acceptable agreement.

We assume a MES system with a finite set of users $U = \{u_1, u_2, \ldots.. u_n\}$ and a finite set of items $I = \{i_1, i_2, \ldots i_m\}$. Each item, in turn, has a finite set of keyword vectors $V = \{v_1, v_2, \ldots v_k\}$ which give more specific details about a movie (for example, story, actor, location, music, action etc.). In our work, we use four keywords for each item. To collect the ratings of each criterion for each item, n matrices are required with each of size m × k. Each entry of a matrix represents the rating of criteria of an item given by a specific user. The weights $W = \{w_1, w_2, \ldots w_k\}$ given by every user to each criterion of the items are stored in a n × k matrix.

This multilateral consensus scheme is modeled as many one-to-one bilateral consensus schemes. Consensus takes place in a discrete number of rounds. In each consensus round, each expert makes an offer in alternate fashion, according to its preferences. If the offers from both experts (one offered and another existing user) overlap in a round, the consensus is successful and both parties reach an agreement. If the offers do not overlap, then the consensus process continues to the next round where the experts may put the next offer or agree on that offer. If there is no consensus after the limit is reached, one of the experts terminates the consensus.

Our proposed MES-GRS-GA consensus scheme is considered as many one-to-one bilateral consensus schemes in which an expert negotiates with each of its opponents and keeps the results isolated. We have applied GA to obtain the maximum utility offer for each expert in the group. At the end of the consensus phase, we have generated the most appropriate rating for each expert in the group. GA has been also employed to produce a list of ratings with the minimum sum of distances in the recommendation generation phase. Basically, a real-world consensus has large and very complicated consensus space with many issues like limited resources, time constraint and information about the opponent's preferences. To deal with these issues, the baseline consensus model needs further improvements. Therefore, we have employed GA based consensus to find optimal solutions (mutually acceptable agreement). The details involved in this work are discussed below.

3.2 Learning Weight by Using Real-Valued GA

Until now, we considered the consensus process with some random weights given by each user to the attributes. Practically, to improve the consensus process, we proposed an offline method to learn these weights. Initially, some random weights are assigned to all features (a range is specified for every weight). Every time crossover and mutation operator generates a new offspring with better fitness. In our GA approach, the chromosome structure is explained in Figure 1 with bit pattern is represented as a set of weights $[w_1, w_2, \ldots w_4]$, where the value of each weight lies between 0 and 1.

Rankings	5	7	2	9
Bit pattern	0 1 0 1	0 1 1 1	0 0 1 0	1 0 0 1

Figure 1: Chromosome Structure.

– ***Crossover and mutation operator***

The crossover operator is used to preserve and combine the best chromosome of the parents from one generation to next to evolve better new solutions [19]. While crossover operates on two parents, mutation modifies the chromosomes locally by changing one or a few of its characters.

Two parent chromosomes are given as

$$X_1 = \{x_{11}, x_{12}, \ldots\ldots, x_{1k}\} \tag{1}$$

$$X_2 = \{x_{21}, x_{22}, \ldots\ldots, x_{2k}\} \tag{2}$$

The *arithmetic crossover operator* produces two new offsprings according to the value of $\tau = U\,(0, 1)$.

$$X_1^{'} = \tau X_1 + (1-\tau)X_2 \tag{3}$$

$$X_2^{'} = \tau X_2 + (1-\tau)X_1 \tag{4}$$

where X'_1 and X'_2 are newly generated offsprings from X_1 and X_2 parent chromosomes.

The *uniform mutation operator* randomly selects one gene x_i and sets it equal to a uniform random number $U\,(a_i, b_i)$ such that

$$x_j^{'} = \begin{cases} U\,(a_i,\, b_i) & if \;\; i = j \\ x_j & otherwise \end{cases} \tag{5}$$

– ***Fitness function***

The fitness function validates the diversion of the process toward its optimization goal by allowing the best individuals to breed, which leads to a recommendation.

Recursively the chromosomes that obtain high fitness value are chosen as a parent in next generation. The fitness function we employ here is the average difference between actual and predicted ratings for all users in training.

$$Fitness = \frac{1}{n_R}\sum_{j=0}^{n_R} |r_j - p(r_j)| \tag{6}$$

3.3 Consensus Phase

In the Consensus phase we use GA to obtain the maximum utility offer for each user and generate the most appropriate rating for each individual in the group.

Step 1. Generation of consensus preference list

Each expert generates the consensus preference list of items according to their rank and weights. Multi-criteria rating systems have more information about the users and items to use in consensus process [38]. So multi-criteria ratings for an item can provide us more precise approximations about the opponent's behavior. Multi-criteria ratings correspond to user preferences for different components of an item, such as story, acting, direction and visuals in our proposed work. So we can find out the preference list of an expert by the rating of different criteria.

$$r_o = W_S r_1 + W_A r_2 + W_D r_3 + W_V r_4 \tag{7}$$

where r_1, r_2, r_3, and r_4 are the ratings and w_S, w_A, w_D, and w_V are weights of story, action, direction and visuals, respectively. These weights represent the priority that the user offers to the criteria while selecting a movie. For instance, in this work, story criterion might have a high priority, i.e. movies with high story ratings are liked on the whole, regardless of other criteria ratings [48]. Each user will calculate the final rating of each item, and the item with maximum rating will be placed on the top of the preference list and so on. An item with highest payoff value is known as the highest utility offer, and it is denoted as $O_{\max}$ for that expert, and the same for its opponent is denoted as $O_{\min}$. We compute the minimum distance offer to both $O_{\max}$ and $O_{\min}$ by applying GA.

Step 2. Computation of minimum distance offer between each pair of expert using GA
To support the consensus process in a real-world scenario, GA is proposed to find the optimal solution. We employ GA to compute the offer which has the minimum distance to both $O_{\max}$ and $O_{\min}$.

- *Chromosome representation.* Here, the chromosome structure is a sequence of binary numbers representing ratings for the attributes of an item, i.e. story, acting, direction and visuals, respectively.
- *Population initialization.* The initial population is randomly generated. Population size will be the number of chromosomes in the population.
- *Genetic operators.* Genetic operators create new solutions; combine them with existing solutions and select between solutions in order to maintain diversity. Crossover and mutation are the most commonly used genetic operators to produce new offspring.
  - (i) *Selection operator.* Roulette wheel selection method, also known as fitness proportionate selection, is used to select the chromosome according to their fitness value. Each member of the population is assigned a roulette wheel whose size is according to its fitness.
  - (ii) *Crossover operators.* For evolving better new offspring in both the consensus phase and recommendation generation phase, the following crossover operators are used.
    - ***(a) Crossover point across ranking list***. In this crossover, two parents X_1 and X_2 are taken, and a number less than or equal to the size of the chromosome is randomly generated. The offspring are generated by exchanging the rankings of the parents after that random point. Assume that randomly generated point is 2 in Figure 2A.
    - ***(b) Crossover point within bit patterns.*** As described in case (a), two parents X_1 and X_2 are taken, and a number less than or equal to the size of the chromosome is randomly generated. Again, a number less than or equal to the size of bit pattern is randomly generated. The offsprings are generated by exchanging the bit patterns of the parents after that random point (only one-bit pattern at a time) as shown in Figure 2B.

Mutation. Mutation operator randomly selects one gene and replaces its value by a randomly chosen value (e.g. in Figure 3, value 6 is replaced by 7).

Replacement and elitism
By the application of genetic operators, the new population is generated by replacement of the individual chromosome in the current population at generation n + 1 to generation n.

Fitness function
In the population in every chromosome, it is indispensable to measure the quality of the chromosome. This is referred to as measuring the fitness of the possible solution represented by that chromosome.

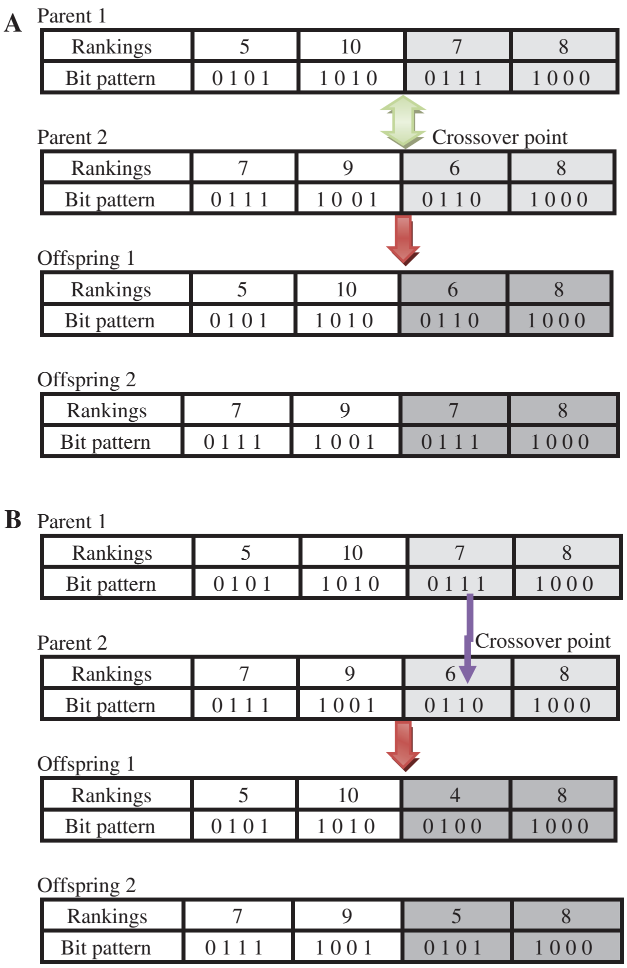

Figure 2: Crossover Operators.
(A) Crossover Point across Ranking List. (B) Crossover Point within Bit Patterns.

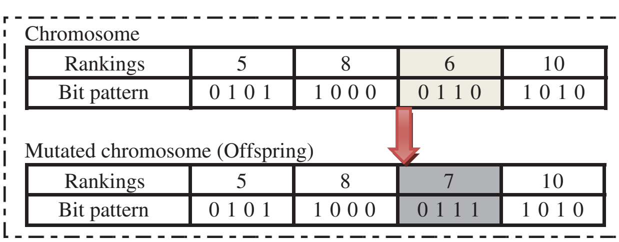

Figure 3: Mutation Operator.

Here the fitness function is *minimum distance offer* which represents the offer having minimum distance to both $O_{\max}$ and $O_{\min}$ and it is calculated by using weighted Euclidian distance formula (as shown in Equation 8), that is,

$$dist(\vec{o}_x, \vec{o}_y) = \sqrt{\sum_{i=1}^{|A|} w_i (d_i^x - d_i^y)^2} \tag{8}$$

where the distance between $O_{\max}$ and *offer* is denoted as D_1 and distance between $O_{\min}$ and *offer* is denoted as D_2. The absolute difference between D_1 and D_2 represent the distance of an offer from $O_{\max}$ and $O_{\min}$.

$$Fitness = abs(D_1 - D_2) \tag{9}$$

Therefore, the optimization problem that proposed scheme aims to resolve the minimization of this function.

Case 1: After applying GA for each pair of experts, if there exists an offer with the minimum distance to both $O_{\max}$ and $O_{\min}$ then store that offer.

Case 2: If there is no such offer then one of the experts (in alternate fashion) propose the new offer with a slight decrement in payoff value and again apply GA to find such an offer for a new pair of offers.

Case 3: After exchanging all the offers, if there is no minimum distance offer and no new offer to propose then end the consensus process without reaching an agreement.

Step 3. Stopping Criteria

GA stops when there is no significant improvement in the fitness value after 10 consecutive generations.

At the end of consensus phase, we have generated the minimum distance offer between two experts. Repeat this process for every pair of experts to find the minimum distance offer for each pair of experts and aggregate the offers to generate the most appropriate ranking list for every individual in the group [32].

3.4 Recommendation Generation Phase

The input to this phase is the most appropriate rank list of every individual, generated in the consensus phase. Here, again we employ GA to find the aggregation list that minimizes the sum of the distance from the input lists. At the end of consensus phase, we have a $n \times k$ matrix where i^{th} row represent the rank of the i^{th} user for k criteria, as shown in Figure 4.

The recommendation generation phase takes place in following steps:

Step 1. Chromosome representation

Here, the chromosome structure is a sequence of rating for the attributes, i.e. story, acting, direction, and visuals, respectively.

Step 2. Initialization of population

In this step, we unsystematically generate initial population, i.e. the number of chromosomes in the population.

Step 3. GAs using genetic operators

The genetic operations are implemented on the current population to generate a new generation.

– ***Selection operator***

The selection operator selects chromosomes from the population on which genetic operators apply. Here, we use roulette wheel selection method.

– ***Crossover operator***

After selection, the crossover is applied on the chromosomes to generate offspring. Here, edge recombination operator (ERO) crossover is used to retain and merge the best characteristics of a parent to generate offspring. The main steps of ERO are following:

(i) Randomly choose the initial number from one of the two parent lists. This number is known as a current number.

(ii) Eliminate all occurrences of current number from the left-hand side in the parent list.

User	v_1	v_2	v_3	v_4
u_1	3	4	1	2
u_2	4	5	2	3

Figure 4: The User Criteria Rating Matrix.

(iii) If there is an entry in the edge list of current number, go to step (iv); otherwise, go to step v.
(iv) The number in the edge list of current number with the lesser number of entries in its edge list becomes a current number. Goto step ii.
(v) If there is any unvisited number, then randomly choose a number and goto step ii. Otherwise, stop the process.

– ***Mutation operator***
Here, scrambled sub-list mutation operator is used to introduce the diversity in the population. A sub-list of numbers is chosen from the parent list and scrambles the list without changing the rest of the chromosome as shown in Figure 5.

Step 4. Fitness function
The fitness function is the *minimum sum of distance offer* which represents the sum of the distance for each individual in the group. We have to find the offer which has a minimum sum of the distance. In order to generate such an offer, the Sum of Kendall tau distance (Sum-KtD) formula is used [6, 40].

$$Sum - KtD(\sigma, r_1, r_2, \ldots, r_n) = \sum_{i=1}^{n} K(\sigma,\ r_i) \tag{10}$$

Now, we have to find out a permutation that minimizes the sum of Kendall tau distance. The KtD is the number of pairs of distinct integers σ_i and σ_j such that $1 \leq i$ and $j \leq n$ where σ_i and σ_j are in opposite orders in each ranking.

– **Stopping criteria**
GA stops when there is no significant improvement in the fitness value after 10 consecutive generations.

At the end of recommendation generation phase, we have generated an aggregation list that minimizes the sum of the distance from the input lists.

3.5 Main Steps of the Proposed Model

The main steps of our proposed MES-GRS-GA are given below:
Step 1: Initially, apply GA to learn the weights of each criterion for every individual in the group.
Step 2: Generate the consensus preference list according to both equal weights (EW) and learned weights (LW).
Step 3: Generate through consensus scheme the most appropriate ranking list for every individual with a maximum payoff in the group by using GA.
Step 4: Apply GA to generate the recommendation list that minimizes the sum of the distances from the input lists.

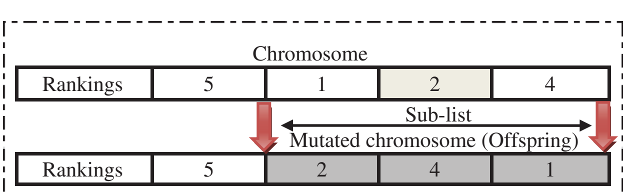

Figure 5: Scrambled Sub-list Mutation Operator.

4 Experimental Evaluation

In this section, we describe the computational experiments performed to evaluate and analyze the effectiveness of the proposed scheme. In Subsection 4.1, we describe the data set used and evolutionary parameter settings. The next subsection shows the effectiveness of proposed GA based consensus scheme (MES-GRS-GA) over baseline consensus scheme [25].

4.1 Experimental Setup

The real data sets that completely match our problem are not publicly available, to the best of our knowledge. Therefore, to evaluate our proposed model, we have conducted experiments on the synthetic data sets that feature close resemblance to a real-world scenario. The characteristics of such simulated networks closely match the real-world settings. We evaluated our proposed model on two data sets.

1. The first consensus profile consists of 10 users and 15 items. Each user gives a rating to all the four keyword vectors of every item. The weights for each criterion take values from the interval of [0 1]. We construct five different consensus groups. In the first consensus group, we generate identical experts (i.e. having the same preferences) and use it as a control group; all other groups are experimental groups. Each successive group is designed with 20% increment in preferential difference.
2. In the second consensus profile, we evaluate the effectiveness of the proposed model when varying the group size and baseline GRSs techniques. It consists of five groups with group sizes equal to 2, 4, 6, 8 and 10.

4.2 Experiments and Results

This section provides details on the results of the experiments performed to evaluate our proposed approach.

4.2.1 Evolutionary Algorithm Parameters

The effectiveness of GA based schemes often depends on the selection of proper values for algorithmic parameters. There are four core parameters (i.e. crossover probability P_c, mutation probability P_m, population size and threshold). The first three parameters are the same for both the phases. Table 1 represents the setting of these parameters to find optimal solutions in all cases.

Three performance measures, joint payoff (JP), success rate (SR) and consensus rounds (CR) are used to measure the effectiveness of the proposed MES-GRS-GA. JP is calculated as the sum of two expert's utilities according to their utility functions at the end of the consensus. SR is defined as the number of cases for which consensus process ends with an agreement. The consensus used is described below:

- MES-GRS-GA-EW: Multi-expert consensus scheme for GRS based on GA with equal weights.
- BCS-EW: Basic consensus scheme with equal weights.
- MES-GRS-GA-LW: Multi-expert consensus scheme for GRS based on GA with learned weights.
- GA-BCS-LW: Basic consensus scheme with learned weights.

Table 1: Criteria of the Evolutionary Algorithm.

Parameter	Value
The probability of crossover P_c	0.8
Probability of mutation P_m	0.2
Size of population	40
Threshold (for consensus phase)	0.0005
Threshold (for recommendation generation phase)	1.0

The following measures are used to estimate the relative efficiency of MES-GRS-GA-EW against BCS-EW:

$$\Delta_{Utility} = \frac{JP_{\text{MES-GRS-GA-}EW} - JP_{BCS\text{-}EW}}{JP_{BCS\text{-}EW}} \times 100\% \quad (11)$$

$$\Delta_{SuccessRate} = \frac{SR_{\text{MES-GRS-GA-}EW} - SR_{BCS\text{-}EW}}{SR_{BCS\text{-}EW}} \times 100\% \quad (12)$$

The measures used to estimate the relative efficiency of MES-GRS-GA-LW against GA-BCS-LW are the following:

$$\Delta_{Utility} = \frac{JP_{\text{MES-GRS-GA-}LW} - JP_{GA\text{-}BCS\text{-}LW}}{JP_{GA\text{-}BCS\text{-}LW}} \times 100\% \quad (13)$$

$$\Delta_{SuccessRate} = \frac{SR_{\text{MES-GRS-GA-}LW} - SR_{GA\text{-}BCS\text{-}LW}}{SR_{GA\text{-}BCS\text{-}LW}} \times 100\% \quad (14)$$

Experiment 1

The purpose of this experiment is to show the effectiveness of GA based multi-expert consensus scheme for GRSs over baseline consensus scheme. The main assumption is that GA based multi-expert consensus scheme for GRSs performs better in a real-world consensus scenario. Here, we also compare the performance of the proposed scheme with EW and LW. First, we compute the average utility and success rate using equal values of weights for the keyword vector. In the next case, a real-valued GA is used to evolve the weights for each user as defined in Subsection 3.3.

Figures 6 and 7 depict the average $\Delta_{Utility}$, $\Delta_{SuccessRate}$, for different consensus groups, and it is clear that in the utility for our scheme, MES-GRS-GA is better than BCS for both equal and LW. A positive value of $\Delta_{Utility}$ and $\Delta_{SuccessRate}$ indicates that the utility and success rate of MES-GRS-GA approach is much more than that of BCS. As depicted in Table 2, the numbers of consensus rounds are much less for our proposed MES-GRS-GA as compared to BCS both for equal and LW. Further, the results show that the proposed scheme with LW consistently outperforms, for all the consensus groups. As expected, for the control group, the performance of MES-GRS-GA and BCS is identical. Indeed, the performance gap increases between these systems as the preferential difference between buyer and seller increases.

Experiment 2

In this experiment the aim is to show the effectiveness of GA based multi-expert consensus scheme for GRSs over baseline consensus scheme. We analyze the effectiveness of the proposed model when varying the group. It consists of five groups with group sizes equal to 2, 4, 6, 8 and 10.

Figures 8 and 9 depict the comparative utility and success rate, respectively, for different group sizes. A positive value of utility and SR denotes that our proposed scheme is better than baseline consensus scheme. Further, the results show that the performance gap increases between these systems as the group size increases.

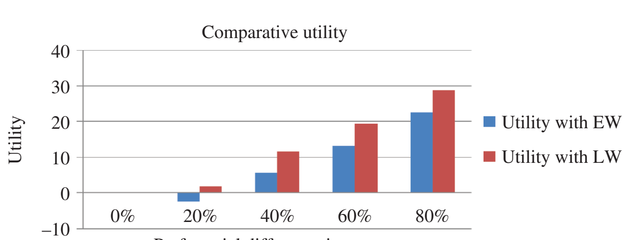

Figure 6: Comparison of $\Delta_{Utility}$ (i) Utility for BCS-EW and MES-GRS-GA-EW; (ii) Utility for GA-BCS-LW and MES-GRS-GA-LW.

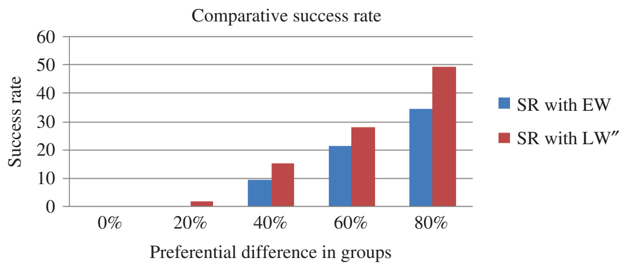

Figure 7: Comparison of $\Delta_{SuccessRate}$ (i) Success Rate for BCS-EW and MES-GRS-GA-EW; (ii) Success Rate for GA-BCS-LW and MES-GRS-GA-LW.

Table 2: Comparison of Consensus Rounds (i) Equal Weights: CR for BCS-EW and MES-GRS-GA-EW (ii) Learned Weights: CR for GA-BCS-LW and MES-GRS-GA-LW.

Group size	Equal weights		Learned weights	
	BCS-EW CR	MES-GRS-GA-EW CR	GA-BCS-LW CR	MES-GRS-GA-LW CR
2	16	8	11	4
4	49	27	33	19
6	71	54	52	35
8	119	98	95	73
10	153	132	138	117

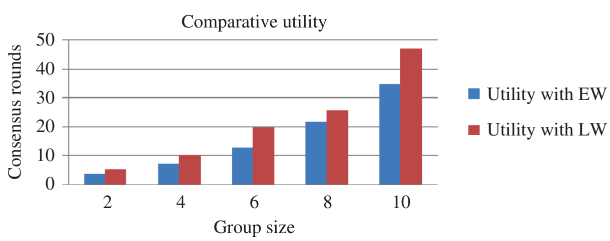

Figure 8: Comparison of $\Delta_{Utility}$ (i) Utility for BCS-EW and MES-GRS-GA-EW; (ii) Utility for GA-BCS-LW and MES-GRS-GA-LW.

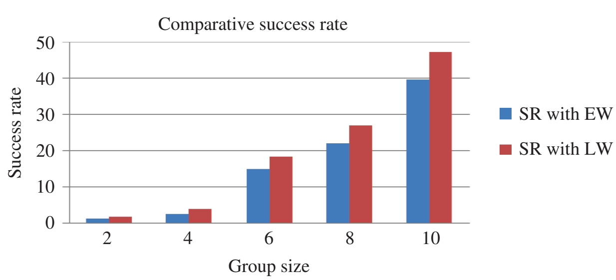

Figure 9: Comparison of $\Delta_{SuccessRate}$ (i) Success Rate for BCS-EW and MES-GRS-GA-EW; (ii) Success Rate for GA-BCS-LW and MES-GRS-GA-LW.

Experiment 3

The results show that the proposed GA based consensus scheme for GRS gives a recommendation list of ratings which maximizes each expert's utility and has the minimum sum of the distance. This experiment shows group satisfaction level (GSL) with EW and with LW for five different group sizes as shown in Figure 10 that clearly indicates that the large size group less satisfy instead of small size group.

We measure the satisfaction level of different groups with EW and LW. GSL is defined as follows:

$$GSL = 1 - (\text{Minimum Sum of Distances})$$

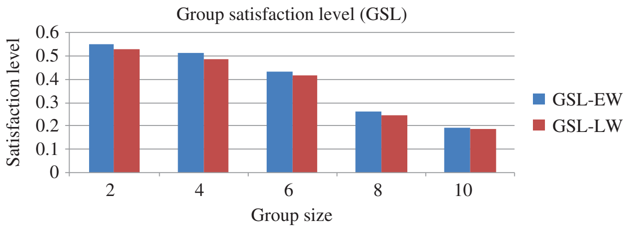

Figure 10: Group Satisfaction Level (GSL) with Equal Weights (GSL-EW) and with Learned Weights (GSL-LW) with Varying Group Size.

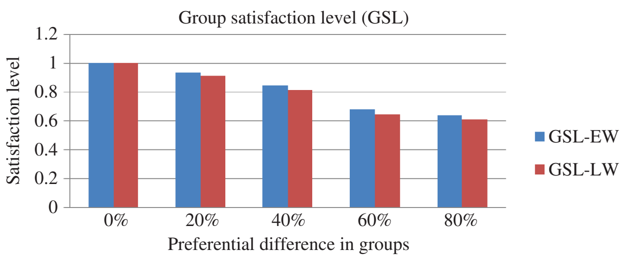

Figure 11: Group Satisfaction Level (GSL) with Equal Weights (GSL-EW) and with Learned Weights (GSL-LW) with Varying Group Size.

It is clear from Table 2 that the satisfaction level of small size groups is higher than that of the large size groups.

It is clear from Figure 11 that the high similarity groups have large satisfaction levels as compared to the groups with distinct preferences, i.e. the identical group has the maximum satisfaction level, and it decreases with increments in preferential difference.

Experiment 4

In this experiment, we have to evaluate the effectiveness of our modal GA based consensus mechanism for a group of users against the baseline techniques of group recommendation. To assess the efficiency of this scheme, we use a standard "IR measure", i.e. "Normalized Discounted Cumulative Gain (nDCG)" [3]. Let p_1, $p_2, \ldots p_k$ be a classified list of elements produced as a group recommendation. Let u be a user and r_{upi} the true user rating u for the most element p_i. The discounted cumulative gain (DCG) and normalized DCG (nDCG) in the k range are, respectively defined as follows:

$$\mathrm{DCG}_k^u = r_{up1} + \sum_{i=2}^{k} \frac{r_{upi}}{log_2(i)} \tag{15}$$

$$\mathrm{nDCG}_k^u = \frac{\mathrm{DCG}_k^u}{\mathrm{IDCG}_k^u} \tag{16}$$

where IDCG is the maximum possible gain value for user u. In Figure 12, the effectiveness of the proposed scheme is compared with baseline GRSs techniques, i.e. LM, MP, average strategy and Borda count which clearly demonstrate that the proposed scheme outperforms all the baseline GRSs techniques.

5 Conclusion

In our work, we have developed GA based framework for the *multi-expert system (MES)* where multiple experts communicate together with the purpose of obtaining a recommendation for the whole. The MES is modeled as many one-to-one bilateral consensuses.

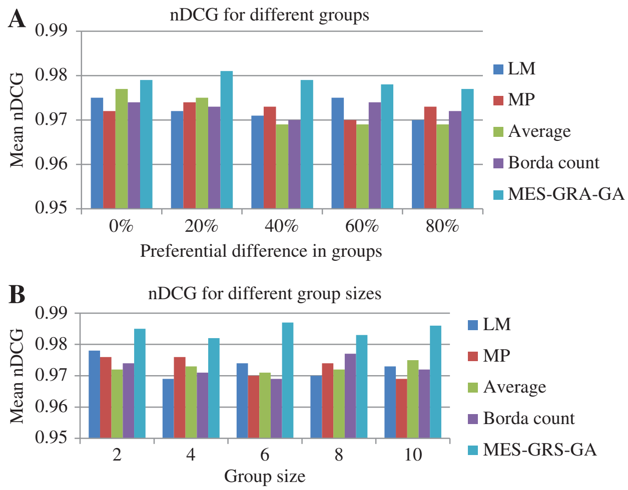

Figure 12: The Effectiveness of the Proposed Model with Baseline GRSs Techniques. (A) The Preferential Difference in Groups. (B) Varying Group Size.

The GA based consensus scheme for GRSs is implemented in two phases. We have applied GA in the consensus phase to generate the offer that maximizes an expert's payoff in the group, whereas in the recommendation generation phase, again we have applied GA to obtain the minimum sum of distances and generated a final recommendation rating list that satisfies all the group members adequately. The results of this work provide experimental evidence that demonstrates the effectiveness of GA based consensus scheme for GRSs in terms of better accuracy. MES-GRS-GA has shown promising results in terms of the satisfaction of group members when compared to different baseline rank aggregation techniques.

6 Future Work

As for future, we would like to extend our work to more complex consensus environments and explore how these changes would affect the agreement. A fascinating future direction would be to assimilate trust-distrust strategies [34, 46] during the consensus process to further improve the proposed scheme. Further, the weights given to each expert can be learned through GA, and its effect on consensus also needs to be investigated.

Acknowledgments: The work presented here has been supported partly by DST-PURSE and partly the RGNF-SRF to the scholar.

Bibliography

[1] S. K. Agarwal and V. Jindal, *MARST: Multi-Agent Recommender System for e-Tourism using reputation based collaborative filtering*, pp. 189–201, Springer-Verlag, New York, Inc., New York, NY, USA, 2014.
[2] L. Ardissono, A. Goy, G. Petrone, M. Segnan and P. Torasso, Intrigue: personalized recommendation of tourist attractions for desktop and handheld devices, *Appl. Artif. Intell.* **17** (2003), 687–714.
[3] L. Baltrunas, T. Makcinskas and F. Ricci, Group recommendations with rank aggregation and collaborative filtering, in: *Proceedings of the fourth ACM conference on Recommender systems*, pp. 119–126, ACM, New York, NY, USA, September, 2010.
[4] P. Bekkerman, S. Kraus and F. Ricci, Applying cooperative negotiation methodology to group recommendation problem, in: *Proceedings of Workshop on Recommender Systems in 17th European Conference on Artificial Intelligence (ECAI 2006)*, pp. 72–75, Riva del Garda, Italy, August, 2006.
[5] Y. Blanco-Fernández, J. J. Pazos-Arias, A. Gil-Solla, M. Ramos-Cabrer, B. Barragáns-Martínez, M. López-Nores, J. García-Duque, A. Fernández-Vilas and R. P. Díaz-Redondo, AVATAR: an advanced multi-agent recommender system of personalized TV contents by semantic reasoning, in: *Web Information Systems WISE*, pp. 415–421, Springer-Verlag, Berlin, 2004.

[6] L. Boratto and S. Carta, State-of-the-art in group recommendation and new approaches for automatic identification of groups, in: *Information Retrieval and Mining in Distributed Environments*, pp. 1–20, Springer, Berlin Heidelberg, 2010.
[7] I. Cantador and P. Castells, Group recommender systems: new perspectives in the social web, in: *Recommender Systems for the Social Web*, pp. 139–157, Springer, Berlin Heidelberg, 2012.
[8] J. Castro, F. J. Quesada, I. Palomares and L. Martínez, A consensus-driven group recommender system, Int. J. Intell. Syst. **30** (2015), 887–906.
[9] J. Castro, J. Lu, G. Zhang, Y. Dong and L. Martinez, Opinion dynamics-based group recommender systems, *IEEE Trans. Syst. Man Cybern. Syst.* **99** (2017), 1–13.
[10] X. Chen, H. Zhang and Y. Dong, The fusion process with heterogeneous preference structures in group decision making, Inform. Fusion **24** (2015), 72–83.
[11] I. Christensen and S. Schiaffino, A hybrid approach for group profiling in recommender systems, *J. Univers. Comput. Sci.* **20** (2014), 507–533.
[12] J. L. De la Rosa, N. Hormazábal, S. Aciar, G. A. Lopardo, A. Trias and M. Montaner, A negotiation-style recommender based on computational ecology in open consensus environments, *IEEE Trans. Ind. Electron.* **58** (2011), 2073–2085.
[13] Y. Dong, Z. Ding, L. Martnez and F. Herrera, Managing consensus based on leadership in opinion dynamics, Inform. Sciences **397** (2017), 187–205.
[14] Y. Dong, M. Zhan, G. Kou, Z. Ding and H. Liang, A survey on the fusion process in opinion dynamics, *Inform. Fusion* **43** (2018), 57–65.
[15] U. Endriss, *Monotonic concession protocols for multilateral negotiation*, pp. 392–399, ACM, New York, NY, USA, 2006.
[16] I. Garcia and L. Sebastia, A negotiation framework for heterogeneous group recommendation, *Expert Syst. Appl.* **41** (2014), 1245–1261.
[17] I. Garcia, L. Sebastia, S. Pajares and E. Onaindia, Approaches to preference elicitation for group recommendation, in: *International Conference on Computational Science and Its Applications*, pp. 547–561, Springer, Berlin Heidelberg, June, 2011.
[18] I. Garcia, S. Pajares, L. Sebastia and E. Onaindia, Preference elicitation techniques for group recommender systems. *Inform. Sciences* **189** (2012), 155–175.
[19] D. E. Goldberg, *Genetic algorithms in search, optimization and machine learning*, Addison-Wesley Publishing Company Inc., Boston, 1989.
[20] S. Ioannidis, S. Muthukrishnan and J. Yan, A consensus-focused group recommender system, *CoRR* abs/arXiv preprint arXiv1312.7076 (2013).
[21] A. Jameson, More than the sum of its members: challenges for group recommender systems, in: *Proceedings of the working conference on advanced visual interfaces*, (AVI '04), pp. 48–54, ACM, New York, NY, USA, May, 2004.
[22] A. Jameson and B. Smyth, Recommendation to groups, in: *The Adaptive Web*, pp. 596–627, Springer, Berlin Heidelberg, 2007.
[23] A. Jameson, S. Baldes and T. Kleinbauer, Enhancing mutual awareness in group recommender systems, in: *Proceedings of the IJCAI*, August, 2003.
[24] J. K. Kim, H. K. Kim, H. Y. Oh and Y. U. Ryu, A group recommendation system for online communities, *Int. J. Inform. Manage.* **30** (2010), 212–219.
[25] R. Y. Lau, M. Tang, O. Wong, S. W. Milliner and Y. P. P. Chen, An evolutionary learning approach for adaptive negotiation agent, *Int. J. Intell. Syst.* **21** (2006), 41–72.
[26] M. Lenar and J. Sobecki, Using recommendation to improve negotiation in agent-based systems, *J. Univers. Comput. Sci.* **13** (2007), 267–286.
[27] B. Li, L. Chen, X. Zhu and C. Zhang, Noisy but non-malicious user detection in social recommender systems. *World Wide Web* **16** (2013), 677–699.
[28] J. S. Lopes, S. Alvarez-Napagao, R. Confalonieri and J. Vázquez-Salceda, *USE: a multi-agent user-driven recommendation system for semantic knowledge extraction*, Technical University of Catalonia, Barcelona, Spain, 2009.
[29] N. Manouselis and C. Costopoulou, Analysis and classification of multi-criteria recommender systems, *World Wide Web* **10** (2007), 415–441.
[30] V. N. Marivate, G. Ssali and T. Marwala, An intelligent multi-agent recommender system for human capacity building, *Electrotechnical Conference, 2008. MELECON 2008. The 14th IEEE Mediterranean. IEEE Xplore* (2008), 909–915.
[31] R. Meena, Group recommender systems – evolutionary approach based on consensus with ties, in: S. Satapathy, J. Tavares, V. Bhateja and J Mohanty, eds., *Information and Decision Sciences. Advances in Intelligent Systems and Computing*, vol. 701, Springer, Singapore, 2018.
[32] R. Meena and K. K. Bharadwaj, Group recommender system based on rank aggregation – an evolutionary approach, in: R. Prasath and T. Kathirvalavakumar, eds., *Mining Intelligence and Knowledge Exploration. Lecture Notes in Computer Science*, vol. 8284, pp. 663–676, Springer International Publishing, Springer, Cham, 2013.
[33] J. Mesthoff, The pursuit of satisfaction: an effective state in group recommender systems, in: *International Conference on User Modeling*, pp. 297–306, Springer, Berlin Heidelberg, July, 2005.
[34] S. Nepal, C. Paris and A. Bouguettaya, Trusting the social web: issues and challenges. *World Wide Web* **18** (2015), 1–7.

[35] A. A. Niknafs and H. Baghche Band, Improved win-win quiescent point algorithm: a recommender system approach, *J. Appl. Sci*. **10** (2010), 3084–3090.

[36] M. O'Connor, D. Cosley, J. A. Konstan and J. Riedl, PolyLens: a recommender system for groups of users, in: *ECSCW 2001*, pp. 199–218. Springer, Netherlands, 2001.

[37] I. Palomares, F. J. Estrella, L. Martinez and F. Herrera, Consensus under a fuzzy context: taxonomy, analysis framework AFRYCA and experimental case of study, *Inform. Fusion* **20** (2014), 252–271.

[38] P. Resnick and H. R. Varian, Recommender systems, *Commun. ACM* **40** (1997), 56–58.

[39] F. Ricci, D. Cavada and Q. N. Nguyen, Integrating travel planning and on-tour support in a case-based recommender system, in: *Proceedings of the Workshop on Mobile Tourism Systems*, pp. 11–16, 2002.

[40] F. Ricci, L. Rokach, B. Shapira and P. Kantor, *Recommender systems handbook*, 1st ed., Springer, Springer-Verlag, Berlin, Heidelberg, 2010.

[41] M. Salamo, K. McCarthy and B. Smyth, Generating recommendations for consensus negotiation in group personalization services, *Pers. Ubiquit. Comput*. **16** (2012), 597–610.

[42] L. Sebastiá, A. Giret and I. García, A multi-agent architecture for single user and group recommendation in the tourism domain, *Int. J. Artif. Intell*. **6** (2011), 161–182.

[43] P. Skocir, L. Marusic, M. Marusic and A. Petric, Agent and multi-agent systems, Technologies and applications, in: *5th KES International Conference, KES-AMSTA 2011*, pp. 104–113, Springer, Manchester, UK, (2001).

[44] E. H. Viedma, F. J. Janusz, J. Kacprzyk and W. Pedrycz, A review of soft consensus models in a fuzzy environment. *Inform. Fusion* **17** (2014), 4–13.

[45] C. Villavicencio, S. Schiaffino, J. A. Diaz-Pace and A. Monteserin, PUMES-GR: a negotiation-based group recommendation system for movies, in: *Advances in Practical Applications of Scalable Multi-expert Systems, The PAAMS Collection. Lecture Notes in Computer Science, vol. 9662*, pp. 294–298, Springer International Publishing, 2016.

[46] Y. Wang, L. Li and G. Liu, Social context-aware trust inference for trust enhancement in a social network based recommendations on service providers, *World Wide Web* **18** (2015), 159–184.

[47] M. Wooldridge, *An introduction to multiexpert systems*, 2nd ed., John Wiley & Sons, Wiley, Chichester, UK, 2009.

[48] W. Zhang, Relational distance-based collaborative filtering for e-learning, in: *Computational Intelligence and Design. ISCID'08. International Symposium on Vol. 2*, pp. 354–357, IEEE, October 2008.

[49] R. Zhang, S. Zhang, S. Ye, Y. Zhao, J. Ford and F. Makedon, Providing recommendations in scene, *Electron. J. E-commerce Tools Appl*. **2** (2009), 9.