Inteligencia Artificial 50(2012), 18-29

INTELIGENCIA ARTIFICIAL

http://erevista.aepia.org/

Ratings Estimation on Group Recommender Systems

Ingrid A. Christensen and Silvia Schiaffino
ISISTAN (CONICET - UNCPBA)
Campus Universitario, Paraje Arroyo Seco, Tandil, Argentina
{ingrid.christensen, silvia.schiaffino}@isistan.unicen.edu.ar

Abstract Recommender systems have been developed to find personalized content for users. The personalization techniques used in these systems, usually focus on satisfying the needs of individual users. Nevertheless, within some domains there is a need to generate recommendations to groups of users instead of individuals. In order to detect the interest of a group of users, several aggregation techniques have been developed. A disadvantage of these techniques is that they require a large amount of computations to estimate unknown ratings. In this article, we present an analysis of the impact of estimating ratings when an aggregation technique is used. For that purpose, we describe a hybrid approach to generate group recommendations based on group modeling. We also present the results obtained when evaluating the approach and two well-known aggregation techniques in the movie domain, and the variations of those results when the estimation process is not included.

Keywords: Group Recommendation, Group Profiling, Aggregate Ratings, Ratings Estimation.

1 Introduction

The growth of the Internet in the last decades has resulted in huge amounts of information. To overcome this problem, personalization techniques have been extensively developed. Recommender systems are one technique for personalization. In these systems, the personalization occurs when the system collects information about users' interests and suggests information based on those interests. Recommender systems have been developed to deal with information overload, finding personalized content for users. As an essential type of information filtering, recommender systems have gained researchers' attention in recent decades and have been successfully introduced in several e-commerce sites, such as Amazon[1] and Netflix[2]. There is extensive research focused on satisfying individual users' needs based on several personalization techniques, namely content-based recommendation [10], collaborative filtering [11] or demographic profile [9]; some agents even combine these techniques to produce hybrid techniques [12].

Within some domains, such as restaurants, TV programs, movies [3], music [3] or tourism [1], activities tend to be social rather than individual, which puts forward the need of adaptation of the classic recommender systems, since the end user of the suggestion is a group formed by individual users with particular preferences. For all intents and purposes, group recommender systems could be classified on two main categories: (1) those which perform an aggregation of individuals' preferences (or ratings) to obtain a group prediction for each candidate item; and (2) those which perform an aggregation of individuals' models into a single group model and generate suggestions based on this model. Some of the techniques applied to aggregate individuals' ratings are multiplication, maximizing average satisfaction and minimizing misery, among others. To create a group model reflecting the preferences of the majority

[1] www.amazon.com
[2] www.netflix.com

ISSN: 1137-3601(print); 1988-3064(on-line)

of the group, the systems aggregate group members' prior preferences. Suggestions are generated for the *"virtual user"* representing the group profile, by applying a classic recommendation technique for individual users.

Some aggregation techniques have been utilized in individual recommender systems to adapt their results to the requirements of group recommendation [6]. For example, in [14] the authors construct a group model by defining a function that minimizes the total distance among individual profiles. The work in [7] presents a method to generate group recommendations that consists of two phases. The first phase includes a filtering method based on the group profile, so as to satisfy most members. The second phase includes a filtering method based on individual profiles, so as to reduce the number of unsatisfied members.

Both approaches, aggregating individual preferences or individual models, require a large amount of estimations for unknown ratings. Essentially, the key problem of estimation processes is that the computational complexity escalates dramatically as the number of unknown ratings increases. Because of that, there is a need to analyze the real impact of the estimation process in the final group satisfaction. In recommendation to groups, unlike individual recommendation, there are some indications about the group interests as a whole which are derived from the individuals' preferences given by the members themselves. Therefore, these known preferences could be used to predict the group preferences without a need for estimation.

This article presents an analysis of the impact of the estimation process on group recommendation. For this purpose, we explain a technique to generate recommendations based on group modeling [2]. As other group profiling techniques, this approach estimates unknown preferences to generate groups recommendations. In order to analyze the importance of estimation in group recommendation, we conducted a set of experiments to compare the effectiveness of the aggregation techniques with and without the estimation process. We analyzed both this approach and two well-known aggregation techniques: maximizing average satisfaction and ensuring some degree of fairness.

The rest of the article is organized as follows: Section 2 explains the estimation problem. Section 3 describes the hybrid group modeling approach. Section 4 describes the experimental results obtained when analyzing the technique. Finally Section 5 presents our conclusions.

2 Ratings Estimation

Most of the existing techniques to generate group recommendations are based on members' given ratings only. To create a list of recommended items for a group, these techniques estimate the ratings for unevaluated items and aggregate these ratings to obtain a single one that applies to the whole group. One of the significant problems in these techniques is insufficient overlap. Most aggregation methods obtain a unique value for each item. If there are few items in common among group members, then the predicted value for the group as a whole would be highly dependent on individual estimations. Aggregation of individuals' preferences approach needs to estimate the unknown members' preferences for each candidate item, i.e., if there are I candidate items and M members, the worst case scenario would be estimated IxM unknown preferences. On the other hand, a group profile may consist of any information deemed relevant at the time of personalizing the system. Most group profiling techniques consider individual evaluations in the model, resulting in a need for estimation. If there are I candidates items, S items included in the group model ($S \epsilon I$) and M members, it would be estimated SxM preferences to complete the group profile and then it would be estimated I-S unknown preferences for the group. The computational complexity of the estimation process depends on the number of group members, the candidate items and the items included in the group profile, for the aggregation model approach.

Individual recommendation also requires estimation for each candidate item for the target user when neighborhood techniques are utilized; however, there is no indication or clue about the unknown preferences that would make the estimation process unnecessary. In group recommendation, known individual preferences could be considered as indications of the preferences of the group as a whole, avoiding the high computational-complexity of estimation processing.

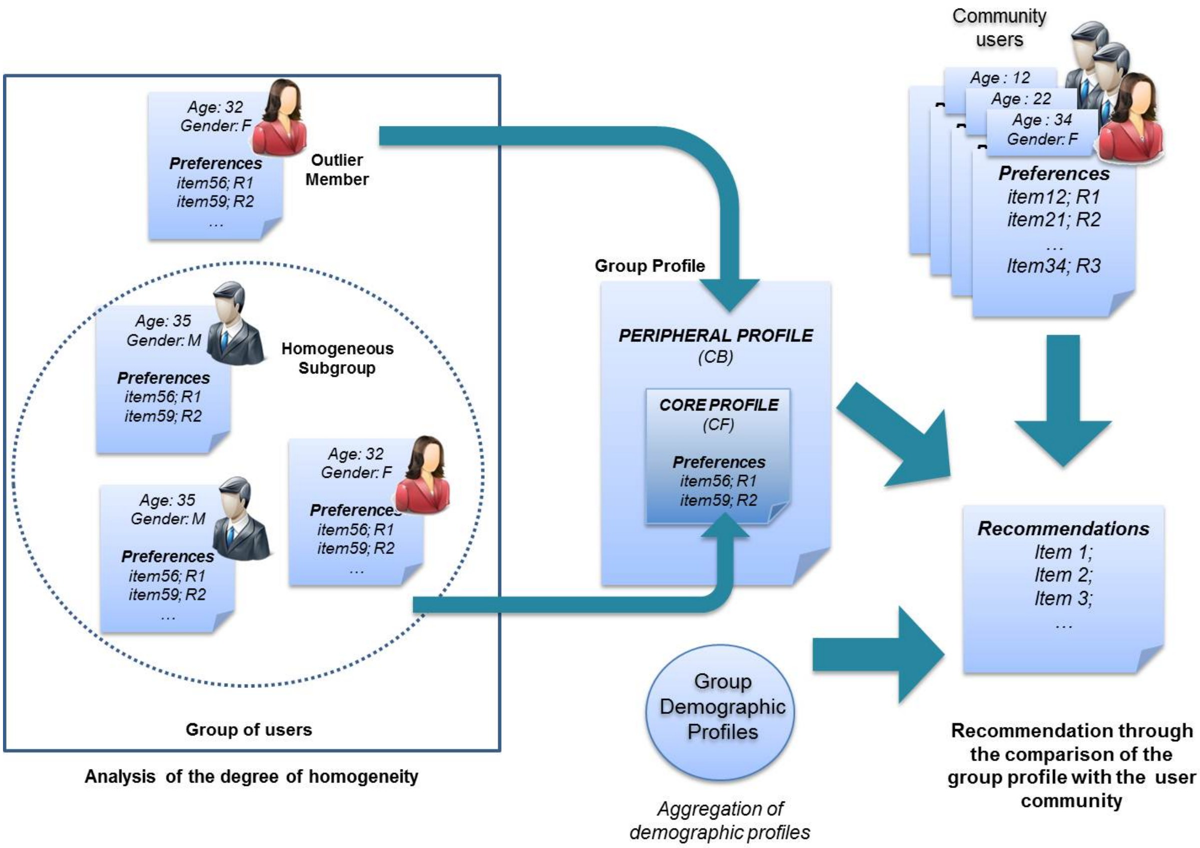


Figure 1: Flow to create the group profile

3 A Group Profiling Approach

The main challenge of the techniques that create group profiles lies in identifying the set of items that should be considered as preferences of the group as a whole. In this work we utilize a hybrid approach to generate group recommendations based on group modeling that considers both homogeneous and non-homogeneous groups. This approach differs from the existing approaches in that it aims at finding implicit similarities between the members' rating profiles, combining three individual recommendation techniques: collaborative filtering, content-based filtering and demographic profile. This combination allows the inclusion of *outliers*[3], by detecting these members with distinctive and/or conflicting preferences, and combining the remaining members in a core homogeneous subgroup. Then the core group profile, with the main subgroup's preferences, is obtained and the outliers' preferences are included (by analyzing the content of the items) to create a profile for the whole group. Additionally, the information included in individual demographic profiles is aggregated to create the demographic profile of the group. To generate the group recommendations, the group profile is contrasted with the community user's profiles, analyzing both rating and demographic profiles. Figure 1 shows the general flow of the technique proposed to create a group profile. The following sections detail the approach.

3.1 Outliers Detection

As mentioned above, first the proposed recommendation process needs to detect the members whose preferences are distant from the majority's. Hence, it is necessary to calculate a cross-correlation of

[3]In statistics, outlier is an observation that is numerically distant from the rest of the data. In this case, we called "outliers" to the members with distant profiles from the rest of the group.

TYPE	DOMAIN APPLICATIONS EXAMPLE	TYPE OF ATTRIBUTE	SIMILARITY MEASURE
Date (Year) Y_n	Release (movie)	$YYYY$	$\frac{(Dif_{max}-\lvert Y_1-Y_2\rvert)}{Dif_{max}}$
String S_n	Editorial (book) - Author (book - music) - Director (movie)	Known set of values (only one)	$S_1 = S_2?1:0$
(String)* S_n^*	Genres (music - movie) - Languages (book - movie) - Actors (movie)	Known set of values (subset)	$\frac{\lvert S_1^* \cap S_2^*\rvert}{S_{max}^*}$
Integer I_n	Years in business (restaurant) - Runtime (music - movie)	Numerical range of values	$\frac{I_{max}-\lvert I_1-I_2\rvert}{I_{max}}$

Table 1: Similarity measures for the attributes types

group members. A confidence factor is included in the correlation calculation, which is determined by the number of overlapping items among user's profiles. Equation 1 calculates the correlation between two users u_i and u_j, where N is the number of overlapping items, r_{max} is the maximum rating domain value (for example, in movie recommendations could be a range that varies between 1 to 5 stars, then r_{max} is 5), $r_{i,x}$ is the rating given by user i to item x, and $r_{j,x}$ the rating given by user j to item x.

$$similarity(u_i, u_j) = \frac{(N * r_{max}) - \sum_{x=1}^{N} |r_{i,x} - r_{j,x}|}{N * r_{max}} \tag{1}$$

To identify outliers we utilize a proximity-based technique introduced by [8]: if m of the k nearest neighbors (where $m < k$) lie within a specific distance threshold d_u then the exemplar is deemed to lie in a sufficiently dense region of the data distribution to be classified as normal. However, if there are less than m neighbors inside the distance threshold then the exemplar is an outlier. With the remaining members, we formed a homogeneous subgroup to construct the *core* group profile. Once the homogeneous subgroup has been formed, the core profile is defined, i.e. the main characteristics of the members belonging to this subgroup are identified. The items included in the profiles of subgroup members become part of the core of the group profile.

3.2 Outlier Inclusion

The content of the items included in the members' profiles is considered to incorporate the outliers. The profile of each outlier is analyzed and items that present higher content similarity than a threshold d_i with the items added to the core profile are included in the *peripheral* group profile. This *peripheral* profile and the *core* profile are then combined to form the rating profile of the group. Item correlation is calculated with equation 2, where N is the number of attribute types, w_x is the weight of the attribute type x, and $f(A_{x,i}, A_{x,j})$ is the similarity between the attribute x for item i and the attribute x for item j. Table 1 shows the similarity equations for the different attribute types considered in this work: *Date*, for attributes that describes years; *String*, for attributes which represent a string holding only one value among several known; *(String)** for attributes that hold a subset of known values; and *Integer*, for attributes that describe a range of numeric values. These equations were adapted from [4].

$$similarity(i_i, i_j) = \sum_{x=1}^{N} w_x * f(A_{x,i}, A_{x,j}) \tag{2}$$

The calculation of item cross-correlation in the core group profile and the items associated to the outliers' profiles allows the inclusion of preferences, not visible a priori, in the user rating profile. These items are included in the group profile with the procedure described in section 3.4.

3.3 Feature Weighting

The item correlation is calculated by analyzing the vectors formed by the values of each attribute (or feature) of the items. These attributes can be numerical or nominal values, such as year, genre, author, among others. A considerable number of techniques aimed at determining the distance between two vectors, such as Euclidean distance or cosine similarity, which consider each attribute equally. However, the human value judgment used to choose an item does not give the same relevance to the different attributes. For example, if a user positively evaluated a book by considering the author more relevant than the year, so it is probable that if he/she selects another book to read, could be interested in those written by the same author rather than those were published in the same year. Therefore, the users base their judgments on some criterion which is a weighted linear combination of differences of individual attributes. Consequently, lets define the linear similarity between two items I_i and I_j, as shown in Equation 3, where w_n is the weight given to the value difference for the attribute A_n among items I_i and I_j. The value difference between each type of attribute is described by $f(A_{n,i}, A_{n,j})$ and is normalized within the range [0,1]. The value of each $f(A_{n,i}, A_{n,j})$ depends on the attribute type (numeric, nominal or boolean) and is given by the functions presented in Table 1.

$$S(I_i, I_j) = w_1 * f(A_{1,i}, A_{1,j}) + w_2 * f(A_{2,i}, A_{2,j}) + ... + w_n * f(A_{n,i}, A_{n,j}) \quad (3)$$

To calculate the similarity between the items included on the core profile and those included on the outliers' individual profiles it is necessary to determine for each attribute weighting values which represent the relevance.

The feature weighting process used in this work is an adaptation of the process presented in [4], in which the weights are derived from a set of linear regression equations. A social network graph is created to reflect the users' criteria to determine the similarity between items. The evaluated items $I_1, I_2, ..., I_n$ are the nodes and the weight of the edges $\#u\{(I_i, I_j)\}$ is the number of users that evaluate each pair of items (I_i, I_j). This value, properly normalized, can be considered as the human value judgment about the differences between I_i and I_j and if we replace in equation 3 this value, we obtain a set of equations (see equation 4) with weights w_n as unknowns. The linear regression equations are derived from this social network graph.

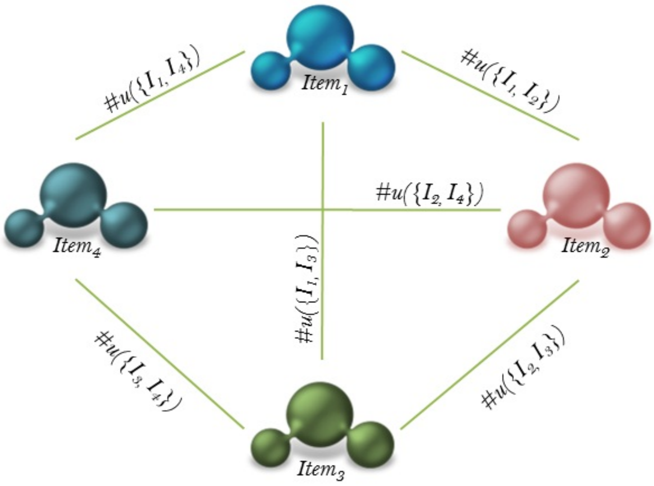
Figure 2 shows an example of a connection graph with four items, which have three different attributes and the six regression equations generated for this graph. In this work we considered as input the set of regression equations and we performed a regression analysis based on SVM (Support vector Machine) technique, which uses the coefficients of the normal vector of a linear SVM as weights of the attributes.

$$w_0 + w_1 * f(A_{1,i}, A_{1,j}) + w_2 * f(A_{2,i}, A_{2,j}) + ... + w_n * f(A_{n,i}, A_{n,j}) = \#u\{(I_i, I_j)\} \quad (4)$$

We focused on the movie domain and considered seven attributes: title, release date, running time, genres, directors, crews, and actors. We implemented this feature weighting process on the movie data set used for the evaluations and we obtained: $w_{title} = 0.121$, $w_{releaseDate} = 0.008$, $w_{runningTime} = 0.39$, $w_{genres} = 0.42$, $w_{directors} = 0.01$, $w_{crews} = 0.001$ and $w_{actors} = 0.05$. These results are presented in [2].

3.4 Group Profile

We applied the well-known rating matrix for individual collaborative recommendations, which represents the users' evaluations of the items (in which the intersection between the row i and the column j contains the evaluation of the user i for the item j). If the cell is empty, it means item j has not been evaluated by user i. In particular, the sub-matrix that includes only group members and the items from both core and peripheral profile are analyzed. Missing evaluations are estimated using a proximity technique including all communities' users with a weighted average. This is the estimation process that is ignored in the experiments to compare the results. The group profile is obtained utilizing combining two well-known aggregation techniques [6]: maximizing average satisfaction and ensuring some degree of fairness. Applying these techniques we obtain a group evaluation R_i for each item, which is composed of a conjunction of the group average and a penalty term that reflects the amount of variation among the predicted ratings. This is represented by equation 5, in which N is the number of group members,



$$
\begin{cases}
w_0 + w_1 * f(A_{1,1}, A_{1,2}) + w_2 * f(A_{2,1}, A_{2,2}) + w_3 * f(A_{3,1}, A_{3,2}) = \#u(\{I_1, I_2\}) \\
w_0 + w_1 * f(A_{1,1}, A_{1,3}) + w_2 * f(A_{2,1}, A_{2,3}) + w_3 * f(A_{3,1}, A_{3,3}) = \#u(\{I_1, I_3\}) \\
w_0 + w_1 * f(A_{1,1}, A_{1,4}) + w_2 * f(A_{2,1}, A_{2,4}) + w_3 * f(A_{3,1}, A_{3,4}) = \#u(\{I_1, I_4\}) \\
w_0 + w_1 * f(A_{1,2}, A_{1,3}) + w_2 * f(A_{2,2}, A_{2,3}) + w_3 * f(A_{3,2}, A_{3,3}) = \#u(\{I_2, I_3\}) \\
w_0 + w_1 * f(A_{1,2}, A_{1,4}) + w_2 * f(A_{2,2}, A_{2,4}) + w_3 * f(A_{3,2}, A_{3,4}) = \#u(\{I_2, I_4\}) \\
w_0 + w_1 * f(A_{1,3}, A_{1,4}) + w_2 * f(A_{2,3}, A_{2,4}) + w_3 * f(A_{3,3}, A_{3,4}) = \#u(\{I_3, I_4\})
\end{cases}
$$

Figure 2: An example graph of correlation between items

i is the item to be evaluated and σ is the standard deviation with a weight w that reflects the relative importance of fairness.

$$R_i = \frac{1}{N} * \sum_{j=1}^{N} r_{i,j} - w * \sigma(\{r_{i,j}\}) \tag{5}$$

Additionally, a group confidence value representing the average of the individual confidences is assigned to each item in the group profile. The individual confidences depend on the source of the ratings, i.e. if they are provided by the user or estimated by the system. If an individual member gives an evaluation to include on group profile, the individual confidence is *1*; alternatively, if the rating is estimated by the system, the individual confidence is a value that represents the incidence of the estimated evaluation calculating the average similarity between the neighbors in the community used for the estimation.

3.5 Generating Group Recommendations

Upon creating this group profile it is possible to generate recommendations with a collaborative filtering technique, looking for users with similar profiles to the target group within the community. The similarity factor is composed by the weighted sum of both collaborative and demographic similarity. Equation 6 aims to calculate this similarity, where α and β are the weight for each similarity ($\alpha > \beta$ and $\alpha + \beta = 1$).

$$similarity(g, u_j) = \alpha * similarity_c(g, u_j) + \beta * similarity_d(g, u_j) \tag{6}$$

Demographic similarity is defined by the users' age and gender. The similarity by age has a maximum value of *0.5* and it is calculated by the normalized difference between the age ranges of user i and j. The similarity by gender is simply *0.5* if both have the same gender; otherwise the demographic similarity would exclusively depend on similarity by age. Furthermore, the collaborative similarity between the group profile and another user within the community is calculated using a variant of equation 1. Besides, the confidence value is considered by weighting each group evaluation, as it determines the quality of the group estimation according to the number of users' evaluations. The predicted group evaluation is obtained by considering the similarity-weighted average of the neighbors' evaluations. The recommendation process concludes with the estimations for each candidate item, suggesting those items with highest estimations.

4 Experimental Results

We carried out two different experiments within the movie domain to analyze the impact of the estimation process on final group satisfaction. In both cases, we utilized the two error metrics most often used in the recommendation literature: *mean absolute error* (*MAE* - equation 7), and *root mean squared error* (*RMSE* - equation 8). Given a test set τ of user-item pairs (*u,i*) with ratings $r_{u,i}$, and the predicted ratings $r_{\bar{u},i}$, *MAE* and *RMSE* determine the error distance between the estimated rating and the real one. *RMSE* penalizes large errors more severely than *MAE*. Since our numerical rating scale gives ratings over the range [1,5], we normalized to express errors as percentages of full scale: *Normalized Mean Absolute Error (NMAE)* and *Normalized Root Mean Squared Error (NRMSE).*

$$MAE = \frac{1}{|\tau|} \sum_{(u,i)\epsilon\tau} |r_{\bar{u},i} - r_{u,i}| \tag{7}$$

$$RMSE = \sqrt{\frac{1}{|\tau|} \sum_{(u,i)\epsilon\tau} (\bar{r}_{u,i} - r_{u,i})^2} \tag{8}$$

Both experiments compare the error produced by aggregation techniques with and without estimation process. In the first experiment we analyzed the prediction for how each member of the group g would rate a subset of items for which the real individual evaluation is known, measuring the individual satisfaction related to the group satisfaction. In this experiment it is analyzed only the hybrid approach since the known individual evaluations are collected from the *test* data set, which is a subset of the whole data set that is not included in the *training* data set; to analyze the aggregation of individuals' preferences techniques it is needed to know at least one individual preferences in order to deduce a group value. In the second experiment we analyzed the prediction for how group g would rate a subset of items for which the real evaluation of the group is known, measuring the group satisfaction as a whole. In order to analyze the impact of estimation in aggregation techniques, we selected maximizing average satisfaction and ensuring some degree of fairness, because some comparative analysis has shown these techniques (and multiplication) are the most successful to achieve individual satisfaction [3]. The goal of maximizing average satisfaction can be achieved by an aggregation function that computes some sort of average of the predicted satisfaction of each member for use as a basis for the selection of candidates. On the other hand, the goal of ensuring fairness is to satisfy everyone just about equally well and is in general combined with some other goal. For example, it could be combined with maximizing average satisfaction with a penalizing term that reflects the amount of variation among the predicted ratings (Equation 5).

4.1 Data Sets

We based the experiments on the Yahoo! Movies Data Set [13] provided for Yahoo! Webscope™ Program[4]. The *training* data contains 7,642 users, 11,915 movies/items, and 211,231 ratings. Moreover, the *test* data contains 2,309 users, 2,380 items, and 10,136 ratings. Besides this, the data set provides complete movie descriptive content information (29 fields per movie). We focused on 7 of them: title, running time, release date, genres, directors, crews and actors.

In order to analyze the group satisfaction on the second experiment, we used the group feedback obtained from a set of 44 System Engineering students at UNCPBA [5]. The students were organized in 9 groups with different sizes (between 3 and 6 users per group). Each group would choose a subset of items, which were used in the second experiment as a real evaluation, which allows us to compare with the evaluation predicted for our approach. These profiles were included as part of the Yahoo! Data Set.

[4]http://research.yahoo.com/Academic_Relations

[5]The student profiles utilized in this experiment are available at: http://users.exa.unicen.edu.ar/~ichriste/projects_en.html

4.2 Experimental Settings

The experiments to analyze the precision of the approach were carried out under a set of assumptions derived directly from the procedure proposed. Firstly, the computation process to obtain the demographic similarity between two users suggests the necessity of sorting out the users' ages in ranges. For the experiments, the users' age was divided in six different ranges: 1) 15 to 24 years old, 2) 25 to 34 years old, 3) 35 to 49 years old, 4) 50 to 64 years old, 5) 65 to 74 years old; and 6) 75 years old or more.

Moreover, the approach applies a neighborhood technique, which requires the definition of the maximum number k of neighbors used for estimation. In the experiments below, we considered k=60, as it is suggested by [5]. Besides, the *outlier* detection process is sensitive to the use of the thresholds of minimum distance d_m between two members to be neighbors and the minimum number m of neighbor members to determine the homogeneous subgroup (if a group member has fewer neighbors than the threshold then it is an *outlier*). In that case, we considered that the minimum distance d_m is dependent on the domain and data. In statistics, a distance value within the range $[media_{s_u} - \theta_{s_u};media_{s_u} + \theta_{s_u}]$ is considered as "*normal*". A value below that range is considered "*outlier*". Therefore, we calculated the mean and the standard deviation (θ_{d_m}) of the distances among all users. We analyzed the cross-correlations among users in the Yahoo! Movie Data Set and we obtained a threshold $d_m = 0.6$. After identifying the value of d_m, we tested the approach by varying the minimum percentage of neighbor members to determine that m representing a 21% have shown acceptable results identifying *outliers*.

Then, we needed to select a value for the threshold d_i that determines the minimum distance between two items for the process to include *outliers*. As for the distance between members, we considered the distance values between items within the range $[media_{s_i} - \theta_{s_i};media_{s_i} + \theta_{s_i}]$ as "*normal*". Hence, we calculated the mean and the standard deviation (θ_{d_i}) of the distances among all items and we obtained a value $d_i = 0.28$.

The aggregation technique used to estimate the evaluation for the items in the group model depends on a weight w which represents the relevance of the standard deviation. If we choose a high value for w, we may obtain a recommendation that makes everyone equally miserable. Because of that, we pick w=*0.1* to give certain relevance to the fairness, but not too much.

The approach proposes a methodology to include the *outliers*, considering the content of the items by weighting the attributes. The weights used in the experiments were empirically evaluated by applying the featuring weighting process presented in the Section 3.2. We defined these weights as follows: $w_{title} = 0.121$, $w_{releaseDate} = 0.008$, $w_{runningTime} = 0.39$, $w_{genres} = 0.42$, $w_{directors} = 0.01$, $w_{crews} = 0.001$ and $w_{actors} = 0.05$. All these values were empirically tested and presented in [2].

4.3 Experiment 1

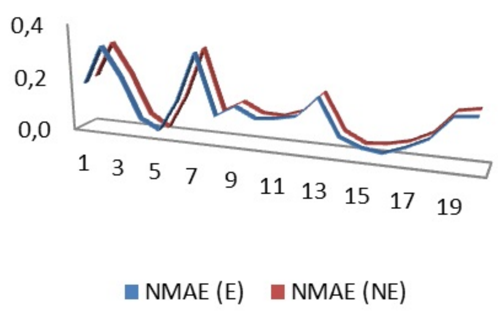
The first experiment aims to compare the *NMAE* and *NRMSE* arose by the group modeling approach including and not including the estimation process, focusing on the individual satisfaction (not group satisfaction). In order to achieve this, we created 20 groups with a total of 118 users from the Yahoo! Data Set and we recommended a set of items included on the *test* data set for each group. The groups were formed with 3 to 9 users. We computed the error metrics for each group member, measuring individual satisfaction (see Figure 3).

4.4 Experiment 2

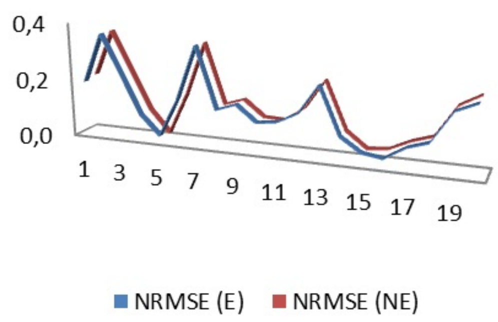
This experiment aims to compare the *NMAE* and *NRMSE* arose by the three aggregation techniques including and not including the estimation process, focusing on group satisfaction. As mentioned above, in this experiment we considered the group feedback obtained from a set of 44 System Engineering students. Figure 4 shows the error metrics values obtained for each group by the three different techniques, using the estimation process in comparison with the aggregation techniques excluding estimations.

4.5 Discussion and Analysis

We compared the prediction values generated for different items using two aggregation techniques and our hybrid approach, in all cases including and excluding the estimation process to obtain the individual



(a) NMAE of Hybrid Approach



(b) NRMSE of Hybrid Approach

Figure 3: Experiment 1 - Individual satisfaction

		Estimation (E)		*no Estimation (nE)*	
		NMAE	NRMSE	NMAE	NRMSE
INDIVIDUAL SATISFACTION	*Hybrid Approach*	0.13	0.15	0.13	0.16
GROUP SATISFACTION	*Ensuring Fairness*	0.21	0.22	0.19	0.26
	Maximizing Average	0.20	0.21	0.21	0.25
	Hybrid Approach	0.19	0.20	0.22	0.25

Table 2: Summarized results

unknown preferences. Results are summarized in Table 2.

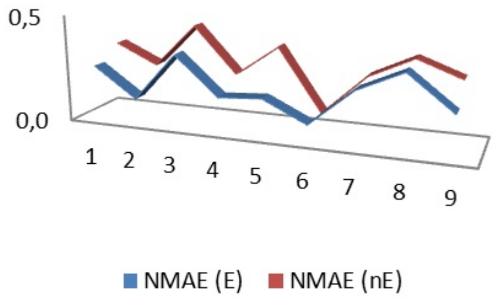
The general results when the estimation process is avoided for each aggregation technique shown that the need to include the estimation process in the aggregation techniques is relative. The first experiment reveals that for the hybrid approach there is no need to include the estimation process when the group profile is created, in both cases the errors (considering the individual satisfaction) are the same. As for the group satisfaction, the second experiment has shown that it would have minimal error differences, considering the high computational complexity which it is demanded for the estimation process. The possibility of avoiding the estimation process could mean a significant reduction in computational complexity of the whole process to generate group recommendations. The hybrid approach requires an offline phase to calculate the individual estimations for each item included in the group profile. If the estimation process is avoided, this phase could be done as part of the online phase. Table 3 shows the time complexity for each aggregation technique analyzed in this work, when the estimations are included and when are excluded. These results were obtained analyzing a group of 4 users of the Yahoo! Data Set, considering 1580 candidates items.

5 Conclusions

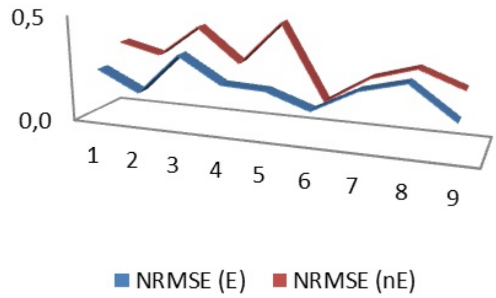
In this paper, we have analyzed the impact of the estimation process in the group satisfaction. The results obtained when evaluating three aggregation techniques demonstrated that it would have minimal error differences when the estimation process is avoided. In conclusion, the estimation process would be

	Hybrid Approach		Ensuring Fairness		Maximizing	
	E	*nE*	*E*	*nE*	*E*	*nE*
Time (ms)	426307	7351	41002	7382	37316	4718

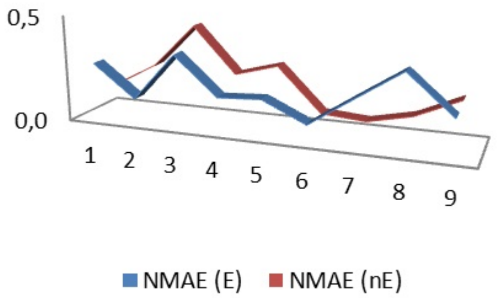
Table 3: Time complexity of the aggregation techniques



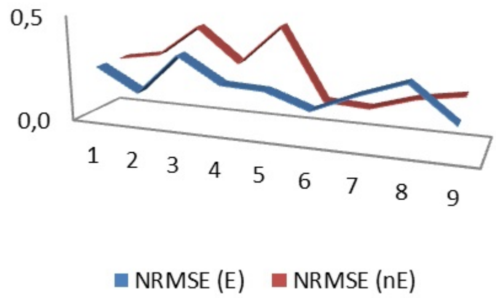
(a) NMAE of Hybrid Approach



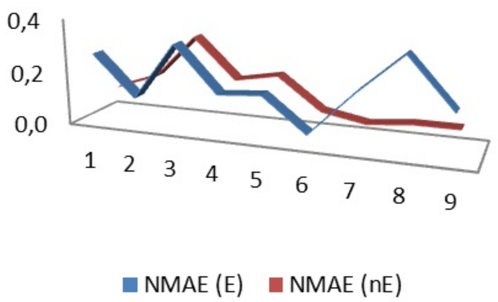
(b) NRMSE of Hybrid Approach



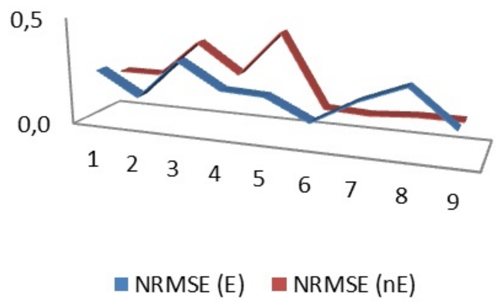
(c) NMAE of Ensuring Some Degree of Fairness



(d) NRMSE of Ensuring Some Degree of Fairness



(e) NMAE of Maximizing Average Satisfaction



(f) NRMSE of Maximizing Average Satisfaction

Figure 4: Experiment 2 - Group satisfaction

necessary when limited information about individual preferences is known; as is the case of the aggregation of individuals' preferences techniques, which require that at least one known individual preference. On the other hand, the aggregation of individuals' models techniques do not have this problem, because they combine the individual models, in which the individuals' preferences are known, and predict from this combination the group evaluation for each candidate. Most of the group recommender systems use aggregation techniques to generate group recommendation that compute a large amount of estimations. As far as we know, there is no previous work in the area that analyzes the real impact of avoiding estimations in the group and individual members' satisfaction.

6 Acknowledgements

This work has been partially funded by ANPCyT, Argentina, through Project PICT-2011-0366.

References

[1] Liliana Ardissono, Anna Goy, Giovanna Petrone, Giovanna Segnan, and Giovanna Torasso. Intrigue: Personalized recommendation of tourist attractions for desktop and handset devices. In *Applied Artificial Intelligence*, pages 687–714. Taylor and Francis, 2003. 1

[2] Ingrid A. Christensen. Un enfoque híbrido para la recomendación a grupos de usuarios. Master's thesis, Facultad de Ciencias Exactas, UNCPBA, December 2011. 1, 3.3, 4.2

[3] Ingrid A. Christensen and Silvia Schiaffino. Entertainment recommender systems for group of users. *Expert Systems with Applications*, 38(11):14127 – 14135, 2011. 1, 4

[4] Souvik Debnath, Niloy Ganguly, and Pabitra Mitra. Feature weighting in content based recommendation system using social network analysis. In *Proceeding of the 17th international conference on World Wide Web*, WWW '08, pages 1041–1042, New York, NY, USA, 2008. ACM. 3.2, 3.3

[5] Jon Herlocker, Joseph A. Konstan, and John Riedl. An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. *Inf. Retr.*, 5:287–310, October 2002. 4.2

[6] Anthony Jameson and Barry Smyth. Recommendation to groups. In *The Adaptive Web: Methods and Strategies of Web Personalization*, chapter 20, pages 596–627. 2007. 1, 3.4

[7] Jae Kyeong Kim, Hyea Kyeong Kim, Hee Young Oh, and Young U Ryu. A group recommendation system for online communities. *International Journal of Information Management*, 30(3):212–219, 2010. 1

[8] Edwin M. Knorr and Raymond T. Ng. Algorithms for mining distance-based outliers in large datasets. In *Proceedings of the 24rd International Conference on Very Large Data Bases*, VLDB '98, pages 392–403, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc. 3.1

[9] Bruce Krulwich. Lifestyle finder: Intelligent user profiling using large-scale demographic data. *AI Magazine*, 18(2):37–45, 1997. 1

[10] Michael Pazzani and Daniel Billsus. Content-based recommendation systems. In Peter Brusilovsky, Alfred Kobsa, and Wolfgang Nejdl, editors, *The Adaptive Web*, volume 4321 of *Lecture Notes in Computer Science*, chapter 10, pages 325–341. Springer Berlin / Heidelberg, Berlin, Heidelberg, 2007. 1

[11] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. pages 285–295, 2001. 1

[12] Silvia Schiaffino and Analía Amandi. Building an expert travel agent as a software agent. *Expert Systems with Applications*, 36(2, Part 1):1291 – 1299, 2009. 1

[13] Yahoo! Academic Relations. R4 - Yahoo! movies user ratings of movies and movie descriptive content information, version 1.0., 2002-2006. 4.1

[14] Zhiwen Yu, Xingshe Zhou, Yanbin Hao, and Jianhua Gu. Tv program recommendation for multiple viewers based on user profile merging. *User Model. User-Adapt. Interact.*, 16(1):63–82, 2006. 1