Group Recommender Systems as a Voting Problem

George Popescu

École Polytechnique Fédérale de Lausanne, Switzerland
Systemic Modeling Laboratory (LAMS), BC 167, 1015, Lausanne, Switzerland
george.popescu@epfl.ch

Abstract. Nowadays, technology allows for a better understanding of user needs through system design (recommender system) methodologies that position the individual at the center of all his actions. In this paper we start by reviewing the state of the art in both individual and group recommender systems technologies. On this ground we cluster the main characteristics of recommender systems with respect to the tasks they perform, the methods they employ and the issues they address. The other theoretical part we rely on is derived from social choice theory and voting. The main objective of this paper is to highlight the role of voting in group recommender systems, more precisely discussing several voting methods together with their characteristics. Our main contributions focus on: reviewing the state of the art literature related to voting in GRS, proposing an innovative and transparent voting mechanism and highlighting the current development of our music recommender system, GroupFun.

Keywords: Behavior, Social choice, Game Theory, Group Decision Making, Incentives, Preference Aggregation, Recommender Systems, Voting.

1 Introduction

Online recommendation technology, for instance, offers the possibility of understanding users' preferences after just a few clicks. In addition, the interaction of various online services can offer a more precise user decision model and propose products that might interest him. Another dimension explored in recommender systems in recent years is the use of social resources to elicit users' preferences and reduce individual effort. Thus, individuals can benefit from excellent recommendations through their network or group of friends. The ubiquitous nature of recommender systems in online commerce websites suggest this new approach helping users make effective decisions, filtering information and allowing companies to increase their revenue through product promotion by targeting an entire group rather than a single, isolated individual.

A (individual) recommender system is a system which, through an information filtering technique, attempts to recommend information items - e.g.: music, movies, TV programs, videos on demand, books, news, images, web pages, research papers etc.) which are likely to be of interest to a single user. In individual recommender systems the more effort a user puts in stating his preferences the more accurate recommendations

A.A. Ozok and P. Zaphiris (Eds.): OCSC/HCII 2013, LNCS 8029, pp. 412–421, 2013.

he will obtain. The challenges associated with recommender systems focus on the lack of data: they need a lot of information to effectively make recommendations. Furthermore, recommender systems are “biased towards the old and have difficulty showing new”: the issue of changing data. Also, users’ preferences change over time. This change cannot be very precisely measured and predicted.

In group recommender systems (GRS) the challenges are a lot more complex: e.g. users do not need to interact with the system more and still obtain group-satisfying recommendations. Their preferences need to be understood by the recommender following social rules. Also, users need to have an incentive for stating their preferences truthfully for the entire group. Research in the game theory field provides mechanisms for truthful preference elicitation. Recommender systems, on the other hand, have been used for solving social choice issues such as: information adaptation, preference aggregation and automated negotiation. They offer the potential for substantially improving preference aggregation and elicitation.

Understanding group recommenders issues relate to the interaction between the system and the users. However, applications of game theoretic methods to group recommenders are still an open research area. The development of such theoretical consideration and applications is a research priority for the social group recommender systems and human computer interaction research fields. Findings in this field are related to truthful preference elicitation, recommendation understanding and user adoption.

The significance of the current article is two folded: on the one hand we discuss theoretical concepts grounded in game theory such as incentives for truthful preference elicitation and voting strategies for influencing a group decision, and, on the other, we showcase on implementation of a truthful voting scheme implemented in our Facebook application. In GroupFun, users can contribute music to their group and rate each other’s songs while seeing others’ ratings. Through this design we can measure the extent to which some users are more individualistic trying to get their songs voted to the top of the playlist whereas others are more group oriented, giving “fair” ratings. The results mentioned in this paper have an impact in dynamic online environments when users vote (and change their votes) numerous times. Instead of competing for the desired outcome we shown how a probabilistic voting scheme can help users state their preferences truthfully. Using a simple algorithm we have defined an incentive-compatible scheme in which scores are interpreted as probabilities.

2 Recommender Systems and User Preferences

The way an individual recommender system works is that typically it compares a user profile to some reference characteristics, and tries to predict the 'rating' that a user would give to an item they had not yet considered based on these characteristics which may belong to the information item (the content-based approach) or the user's social environment (the collaborative filtering approach) (McCarthy et al. 1998).

A group recommender system is a recommender system aimed at generating a set of recommendations that will satisfy a group of users, with potentially competing interests. The challenges associated with this simple statement deal with: considering how to record and combine the preferences of many different users as they engage in simultaneous recommendation dialogs (Jameson, 2004).

Recommender systems are widely extended in most online applications and platforms. They help users reach the items they want instead of searching them online by finding out which might be of interest to the user. Social networks can be of help in the sense that one's friends can decide which items may be recommended for someone else through means of sharing the same interests. Furthermore expertise can be both useful and not: sometimes the experts' approvals do not reach the mass. From an economic point of view 2/3 of Netflix rented movies are due to recommendations, 38% of Google News clicks are due to recommendations and 35% of Amazon's sales are due to them also (Adomavicius et al. 2010). On the internet everything can be recommended under the generic name of "item": music, books, news, advertisements, cloths, programming code, friends, cafes, restaurants, etcSocial Choice Theory

Social choice theory is a theoretical framework for measuring individual interests or welfares as an aggregate towards collective decision (Chevaleyre et al. 2007). Social choice theory and decision-making theory are strongly connected. Much advancement in both fields contributes to the success of the other field: social choice deals with evaluation of methods for collective decision-making while decision making helps putting decisions into practice for maximizing social welfare. When extending individual decision-making or set of preferences to a group or collective decision making process one needs to take into account far more preference levels and intensities related to the field of choice in order to maximize a common welfare state, payoff or satisfaction function (Gruenfeld 2006, Hastie and Kameda 2005).

The main challenges discussed in the social choice literature are related to: social filtering, group formation, strategy-proofness, unconditional privacy, satisfaction measurement (Masthoff 2005, Masthoff 2006), coalition formation, recommendation collaboration and negotiation (Chevaleyre et al. 2007).

Jameson (2004) explores the challenges for group recommender systems while finding the response to four novel research issues: "What benefits and drawbacks can member preference specification have, and how can it be supported by the recommender system?", "How can the aggregation procedure effectively discourage manipulative preference specification?", "How can relevant information about suitability for individual members be presented effectively?" and "How can the system support the process of arriving at a final decision when members cannot engage in face−to−face discussion?"

Jennings et al. (2001) examine the space of negotiation opportunities for autonomous agents with the purpose of defining automated negotiation prospects, methods and challenges Negotiation is presented as the best method for management and resource sharing in a multi-agent environment. It also evaluates negotiation key techniques and presents some major challenges for future automated negotiation research. The general negotiation framework is modeled on the basis of agreements and proposals. The space between negotiation acceptance (agreement) and refusal opens the discussion for efficiency and effectiveness.

3 Voting in Recommender Systems

Recommender systems have emerged in the mid-1990s when forecasting theories and information retrieval algorithms have linked this domain with social choice modeling. The most basic formulation of a recommender system is that of aggregating a set of user preferences – which can be either implicit or explicit – into a common social welfare function which would maximize the satisfaction of all users. For online applications especially voting is one of the most common used ways for users to manifest their preferences. Ratings can come after users' interaction with the system or by interpreting his/her preferences as extract of personal data. Once votes are submitted the recommender system should come up with solution corresponding to the highest scored items. The difficulties encountered by such a system are numerous: the number of users, their preferences, cold start or initial recommendation, complex interpretation of preferences, utility statements, fact and desirability, etc. Voting difficulties relate back to: number of votes, user interaction, voting scale, which items to be displayed first, voting estimation for non-voted items, voting differences in interpretation across users, lack of an absolute framework, etc.

In recommender systems utility is typically represented through users' votes or ratings. The central problem of voting in recommender systems is connecting rated items with unrated ones. Through their nature, group recommendation systems aim at recommending items that are most relevant for the common interest of a group of users. In most cases voting mechanisms assume that users rate all (or some) items in order to identify the item (or a group if items) that suits the preferences of all group members. This represents a very strong assumption since it is mostly desirable that users should do the least of effort while expecting that the recommender system will know their common preferences. So the above assumption proves as not being feasible in sparse rating scenarios which are very common in the field of recommender systems. Compared with other decision mechanisms such as negotiations, coalitions and actions, voting is a very common and easy framework for helping users reach a common output. It becomes desirable to determine the winning item(s) while using a minimal set of the group members' ratings, under certain assumptions of the voting mechanism of the recommender system. Voting can be a very computationally costly mechanism thus yielding the need for effectiveness. Heuristic algorithms prove to be extremely useful in scenarios depending on user interest and their effort: minimizing the number of user required ratings, for instance.

4 GROUPFUN

GroupFun is a Facebook application available at the address http://apps.facebook.com/groupfun/ and hosted at EPFL.

Fig. 1. "Party list" tab in GroupFun

4.1 Home

The “Home” page contains the visual identify of the GroupFun and three playlists: Top 8 GroupFun, Christmas and Lausanne Party. Three entities are samples of what GroupFun can have as output, as shown above.

4.2 My List

Users can create their own playlist from a number of 10.000 songs. After the playlist is created, the user can rate the songs, as in the figure 2. The music player, soundmanager, can help the user to take the right decisions. The user can edit his/her playlist and add/remove songs from the playlist.

4.3 My Friends

The user can invite his/her friends to use the application and check their activity: they accepted or not the invitation and what are their music preferences. In the implementation, we used the standard Facebook request fb:multi-friend-selector, customized with 6 maximum invitations and 5 friends per column. The activity of user's friends is available, in case that he/she wants to check their music preferences. This feature increases the interaction within a group of friends, as some users can rate the songs already rated in the system. A preview is available in the figures below.

4.4 My Scrobbler

Using the Last.fm music recommender system called “Audioscrobbler”, we imported users data into GroupFun by taking advantage of the profile of each user's musical taste after recording details of the songs the user listens to, either from Internet radio stations, or the user's computer or many portable music devices. This information is transferred to Last.fm's database (“scrobbled”) and then scrobbled again into GroupFun. The profile data is then displayed on the user's profile page.

4.5 Party List

Users can express their preference related to the songs from the event playlist. For the “Party list” page, we implemented two recommendation algorithms and the output is a common playlist, based on the preferences of all the users.

5 Voting Mechanism

The motivation of this research was to find a preference elicitation and aggregation method for a group deciding on a joint outcome. Two criteria are important for developing this method: it must maximize the group satisfaction, and it must encourage users to state their preferences truthfully so that group satisfaction actually corresponds to satisfaction of user preferences. This problem is a general instance of social choice and often modeled as a voting problem. We let A be the set of all users and S the set of all possible outcomes that can be rated. In a group music recommendation setting, the outcomes are songs si to be selected in a joint playlist. We let each user aj submit a numerical vote score(si,aj) for each song si that reflects its preference for that song. These votes are given as ratings, for example 4 out of 5 stars, and normalized so that the scores given by each user sum to 1. We then assign a joint score to each song that is computed as the sum of the scores given by the individual users:

$$score(s_i) = \sum_{aj \in A} score(s_i, a_j) \tag{1}$$

To choose the songs to be included in a playlist of length k, a deterministic method is to choose the k songs with the highest joint rating. This is a generalized plurality rule. However, this method is not truthful: consider a user who very much likes a song X that is certainly not liked by anyone else, but also has a second best song Y that many others like. This user has no interest to give a high vote for X, since X will never make it into the k songs with the highest joint score. Instead, she should give a stronger vote to Y, which actually has a chance to make it into the selection. In fact, a famous result in game theory, the Gibbard-Satterthwaite theorem, shows that there does not exist a truthful deterministic voting method. However, we note that this theorem does not apply to non-deterministic methods where the choice includes a random element. Consider for example the random dictator rule: we randomly pick one of the users with equal probability, and let this user decide the next song to be chosen. Once chosen, the user knows that her choice will be included, so she will report it truthfully even if it is not very popular among other users.

5.1 Voting Algorithm

As a method for choosing a joint playlist, we thus propose a method we call the probabilistic weighted sum (PWS) that is equivalent to the random dictator rule: we iteratively choose each of the k songs randomly according to the probability distribution:

$$p(s_i) = {score(si)} \Big/ {\sum_{si \in S} score(si)} \tag{2}$$

To illustrate how PWS works, we consider the following example. In the table below, user1, user2, and user 3 represent group members. The score distribution for each of the candidates is displayed in the respective row, and the joint scores are shown below.

Table 1. An example of three users and 6 candidates using PWS

User1	Song 1: 0.1	Song 2: 0.3	Song 3: 0.2	Song 4: 0.1	Song 5: 0.1	Song 6: 0.2
User2	Song 1: __	Song 2: 0.1	Song 3: __	Song 4: __	Song 5: 0.4	Song 6: 0.5
User3	Song 1: 0.4	Song 2: 0.2	Song 3: __	Song 4: 0.2	Song 5: __	Song 6: 0.2
Total score	Song 1: 0.5	Song 2: 0.6	Song 3: 0.2	Song 4: 0.3	Song 5: 0.5	Song 6: 0.9

For a playlist of size 2, the plurality rule would always choose songs 6 and 2, and User3, who prefers song 1, would have no interest to vote for that song. After normalizing the total scores by the sum of the scores, we obtain the following probability distribution for the set of outcomes.

Probability	Song 1: 0.16	Song 2: 0.2	Song 3: 0.1	Song 4: 0.06	Song 5: 0.16	Song 6: 0.3

It would choose the playlist by choosing one song after another using this probabilistic distribution. Compared to other social choice based algorithms, PWS is incentive compatible. That is, it is to the best interest of the individual to reveal his/her preferences truthfully. It is in fact equivalent to a random dictator method, where the dictator will choose a song randomly with the probabilities given by its degree of preference – a reasonable method since nobody wants to hear the same song over and over again. This is because the probability of a song si to be chosen can be written as:

$$p(s_i) = {score(s_i)} \Big/ {|A|} = \sum_{aj \in A} \frac{1}{|A|} score(s_i, a_j) \tag{3}$$

or, in other words, the probability of choosing user aj times the normalized score that user aj has given to song si. And indeed, User3's preference for song 1 yields a significant probability that this song will be included in the playlist.

Advantages of PWS Compared with Other Methods

- Users are free to choose as many or as few songs as they like
- Users can easily rate each song and the system will turn it to utility score
- Ratings are updated permanently

- The algorithm is computationally simple
- Incentive-compatible truthful property is observed

Disadvantages

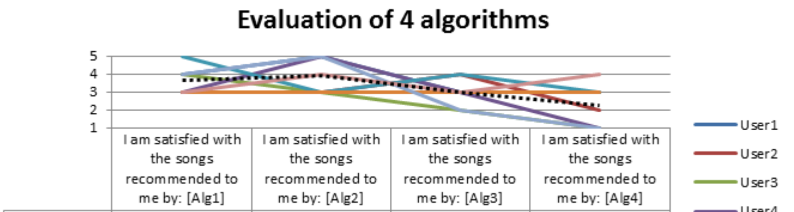
- Difficult to quantify rating differences between distinct users. The weights given by each user cannot be compared with the ones given by another.
- Self-selection effect: most popular songs will receive most votes (not ideal if long tail distribution is desired).

5.2 Evaluation

More than 100 users have tested our system during various pilot tests and experiments organized by uploading and rating music in a small group. Here we report only the results concerning the role of the probabilistic weighted sum algorithm in music recommendation. We planned and carried out an experiment in which 24 individuals evaluated 4 algorithms under the name of Alg1 (Deterministic Weighted Sum), Alg2 (Probabilistic Weighted Sum), Alg3 (Least Misery) and Alg4 (Probabilistic selection).

The results highlighted in the chart and table below together with users comments and inputs are very fruitful for our development of the probabilistic weighted sum algorithm. With colored lines are presented all of the users' ratings (due to space concerns we include only 8 users' ratings) and with a dashed black line the average of all results (16 recordings). We notice a favorable trend for the first two algorithms. The least misery one is less preferred in general by all members compared with the first two whereas the random or probabilistic selection received the least scores.

The last row in the table shows that the average scores for PWS and DWS are very close: 3.625 for the first one and 3.875 for the second one. Given the fact that in our



	I am satisfied with the songs recommended to me by: [Alg1]	I am satisfied with the songs recommended to me by: [Alg2]	I am satisfied with the songs recommended to me by: [Alg3]	I am satisfied with the songs recommended to me by: [Alg4]
User1	4	5	3	3
User2	3	3	4	2
User3	4	3	2	1
User4	3	5	3	1
User5	5	3	4	3
User6	3	3	3	3
User7	4	5	2	1
User8	3	4	3	4
Average	3.625	3.875	3	2.25

Fig. 2. Evaluation of algorithms

experiment users did not have the time to experience the advantages of PWS in many voting sessions we find this result very encouraging for future research. Moreover, we note that none of the users gave a score lower than 3 (out of 5) for PWS and DWS. Other 2 algorithms received very low ratings. Out of the reasons mentioned by our users favoring PWS we report: serendipitious and non-popular music recommendations, transparent information of other members' ratings and discovery effects.

6 Conclusions and Future Work

Recommender systems have known significant improvement in the past years. Many of them have been due to voting protocols and the fact that the system could understand better users' preferences. The work proposed in this paper is strongly connected to dynamic online environments in which users vote and change their votes numerous times. Instead of competing for the desired outcome we have shown how a probabilistic voting scheme can help users state their preferences truthfully. Using a simple algorithm we have defined an incentive-compatible scheme in which scores are interpreted as probabilities. The static and the dynamic cases further contributed to measuring user preference for the deterministic case.

This advances previous work carried on for understanding the voting mechanism as well as its dynamics and user choice. Users are free to state their preferences individually as well as modify them according to some group dynamics factor and intermediate common decision. In real-life examples the two cases presented are very frequently encountered and numerous applications stated in the beginning denote the need for adaptive group recommender systems. GroupFun is one of these systems designed for users to spend the least amount of time stating their preferences and be able to reach the common music playlist goal.

Future work will consider best ways for allowing group members to interactively achieve common outcomes that they are willing to consume. By studying user interaction in a group recommender system we will be able to match group dynamics with music preferences and group satisfaction for a set of events. Fairness is another study point worth investigating in the evaluation. Furthermore we plan to deduct what inspirational process produces user motivation for deciding upon a specific playlist.

References

1. Adomavicius, G., Bockstedt, J., Curley, S., Zhang, J.: Impact of Recommender Systems on Consumer Preferences: A Study of Anchoring Effects (2010)
2. Boyd, D.M., Ellison, N.B.: Social Network Sites: Definition, History, and Scholarship. Journal of Computer-Mediated Communication 13, 210–230 (2008)
3. Chevaleyre, Y., Endriss, U., Lang, J., Maudet, N.: A Short Introduction to Computational Social Choice. In: van Leeuwen, J., Italiano, G.F., van der Hoek, W., Meinel, C., Sack, H., Plášil, F. (eds.) SOFSEM 2007. LNCS, vol. 4362, pp. 51–69. Springer, Heidelberg (2007)

4. Crossen, A., Budzik, J., Hammond, K.J.: Flytrap: Intelligent Group Music Recommendation. In: Proceedings of the 7th International Conference on Intelligent User Interfaces, Intelligent Information Laboratory, Northwestern University, USA (2002)
5. Gruenfeld, D.H.: Group composition and decision making: How member familiarity and information distribution affect process and performance. Organizational Behavior and Human Decision Processes 67(1) (2006)
6. Hastie, R., Kameda, T.: The robust beauty of majority rules in group decisions. Psychological Review 112, 494–508 (2005)
7. Jameson, A.: More than the sum of its members: Challenges for group recommender systems. In: Proceedings of the International Working Conference on Advanced Visual Interfaces, pp. 48–54 (2004)
8. Jennings, N., Faratin, P., Lomuscio, A., Parsons, S., Sierra, C., Woodridge, M.: Automated negotiation: Prospects, methods and challenges. International Journal of Group Decision and Negotiation 10(2), 199–215 (2001)
9. Kay, J., Niu, W.: Adapting information delivery to groups of people. In: Proceedings of the First International Workshop on New Technologies for Personalized Information Access at the Tenth International Conference on User Modeling, Edinburgh (2005)
10. Kerr, N.L.: Group performance and decision making. Annual Reviews Psychology 55, 623–655 (2004)
11. Lang, J.: Some Representation and Computational Issues in Social Choice. In: Godo, L. (ed.) ECSQARU 2005. LNCS (LNAI), vol. 3571, pp. 15–26. Springer, Heidelberg (2005)
12. Masthoff, J.: Group modeling Selecting a sequence of television items to suit a group of viewers (2004)
13. Masthoff, J.: The pursuit of satisfaction Affective state in group recommender systems. In: Ardissono, L., Brna, P., Mitrović, A. (eds.) UM 2005. LNCS (LNAI), vol. 3538, pp. 297–306. Springer, Heidelberg (2005)
14. McCarthy, J.F.: Pocket Restaurant Finder: A Situated Recommender System for Groups. Accenture Technology Labs (2002)
15. McCarthy, J.F., Anagnost, T.D.: MUSICFX: An Arbiter of Group Preferences for Computer Supported Collaboration. In: Proceedings of the 1998 ACM Conference on Computer Supported Cooperative Work. Center for Strategic Technology Research, Accenture (1998)
16. Pennock, D.M., Horvitz, E., Giles, C.L.: Social Choice Theory and Recommender Systems: Analysis of the Axiomatic Foundations of Collaborative Filtering. In: Proceedings of the Seventeenth National Conference on Artificial Intelligence, pp. 729–734. AAAI Press (2000)
17. Ricci, F.: Decision making and recommender systems (2009)