Improving group recommendation with outlier data filtering

Jorge Castro and Manuel J. Barranco and Luis Martínez

Universidad de Jaén, Campus Las Lagunillas, 23071, Jaén
{jcastro,barranco,martin}@ujaen,es

Keywords: Group recommendation; Heterogeneous groups; Disagreement.

New trends in recommender systems face new challenges as group recommendation, in which users give their preferences over items and the system provides recommendations for a group of known users. In certain types of groups, it often occurs that several members do not agree on their preferences over some items so their inclusion in the group recommender system (GRS) may mislead the recommendation results. In this contribution a technique to detect and filter conflictive ratings before their use in the recommendation process is proposed and then its performance evaluated by using a well known recommendation dataset. The results show that rating filtering leads to improvements on GRSs performance.

1. Introduction

A recommender system[1](RS) is a tool that helps users on situations where an overwhelming amount of choices exists and there is no possibility of examining all of them to pick the best one in a reasonable time. Hereby, a RS tries to filter the possible choices by using a set of items for which customer already tried and provided his/her preference about them, trying to predict the best items fitting his/her current needs.

There exist many recommendation techniques,[2] but a simple, effective and widespread technique is collaborative filtering with k-nearest neighbors (kNN-CF). In kNN-CF the recommendations are computed by finding similar users (neighborhood) to the target user and combine their ratings to compute a prediction for the items that the target user did not experience yet, then the top-n items are recommended (see Fig. 1).

Among the new trends in RS,[2] such as context awareness, multiple dimensions recomendations, natural noise, etc; we focus our research on group recommender systems[3] (GRS) which look for suitable recommendations for groups of users (related or not). Usually GRSs suggest products whose purchase or use have a social component to be enjoyed by several people together, such as watch a movie,[1] listen to music[4] or travelling.[5]

Group recommendations are specially challenging in random groups whose members could have different opinions/preferences over the products. This contribution proposes a group recommender technique that pre-filters conflictive opinions in the group for improving group satisfaction regarding the recommendations.

The contribution is structured as follows: section 2 reviews concepts on GRS, section 3 describes the proposed technique for GRS, section 4 shows a case study and section 5 concludes the contribution.

2. Group recommender systems

This section explains the basic concepts on GRS, describing the inputs and basic techniques for group recommendation. Most of RSs use three types of information: users' data ($U = \{u_1, \ldots, u_n\}$), products' data ($I = \{i_1, \ldots, i_m\}$) and users' ratings over the products, to describe how satisfied is a user regarding a particular item ($R \subseteq U \times I \rightarrow D$, D rating domain).

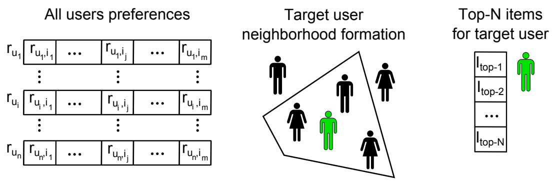

Fig. 1. Single user recommendation kNN-CF.

RSs try to predict ratings for unrated items to perform recommendation using these data. GRS extends RS such that, instead of recommending to one user, recommendations are targeted to groups of users ($G = \{m_1, \ldots, m_r\} \subseteq U$). There exists different modes of group recommendation, such as recommending groups to a user for joining[6] or finding the most suitable group of users for a target item,[7] but we focus on recommending items to a target group of users. Formally, group recommendation consists on finding the item (or set of items) that maximizes the rating prediction for the group of users:

$$GroupRecommendation(I, G) = \arg\max_{i_j \in I}[Prediction(i_j, G)] \qquad (1)$$

There are two basic tecnhiques[8] for GRS: (i) *model aggregation*,[4] which consists on aggregating individual ratings of each member to compute an aggregated group rating profile and perform individual recommendation for this *pseudo-user*; and (ii) *prediction aggregation*,[1,9,10] which computes the list of recommendations for each member and aggregates them into a single group recommendation list (see Fig. 2).

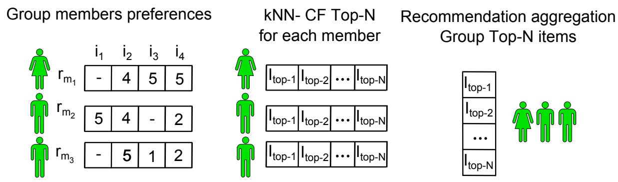

Fig. 2. Group recommendation with prediction aggregation.

Both techniques can aggregate information about a particular item, in which one or several members of the group fully disagreed. In this situation, the group's recommendations are biased negatively.

3. Filtering dissagreements in group recommender systems

In this section, a novel GRS technique is introduced, which filters out items with a high level of disagreement among group members to avoid the biased recommendations aforementioned. Therefore, the application of a filtering process to eliminate disagreement on members ratings might help to improve the recommendation process.

Even though there are different situations for disagreements this contribution, due to page limit, is focused on the following situation: *groups whose members agreed in most product ratings but disagreed in a small set of items*. In this situation the use of data about disagreed items can highly vary the recommendations.

The proposed method is structured in two phases:

(1) The dissagreement of each member rating (see Equation 2) is computed.

$$Disagreement(r_{u,i}) = |\overline{\{R_{G-\{u\},i}\}} - r_{u,i}|,\ u \in G \tag{2}$$

where $r_{u,i}$ is the rating of user u over item i, $R_{G-\{u\},i}$ is the set of ratings from group G members over the same item i without $r_{u,i}$.

(2) Members ratings with dissagreement greater than certain threshold μ are removed from the data used afterwards to compute the group recommendations.

Several special cases should be considered in this situation:

(a) Items rated by $g \leq n$ members: they should not be filtered out. This case is shown in Fig. 3, i_1.
(b) Items with fully disagreement of all members: all their ratings over the same item can be removed (remove item from group data) or keep an aggregated rating for the item. This is controlled by parameter Keep One Rating (KR). This case is shown in Fig. 3, i_3.
(c) Avoid group data deletion: a Maximum Percentage of Deletions (MPD) should be fixed or avoid this technique in datasets whose members are totally different.

Therefore, the proposed technique avoids controversial items, specially when KR parameter is set to $false$. This way, the filtering technique may reduce item coverage in order to gain prediction accuracy.

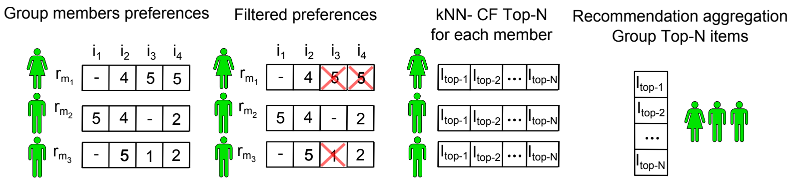

Fig. 3. Group recommendation with prediction aggregation and filtering.

4. Evaluation and results

To validate the proposed method, memory based kNN-CF algorithm with prediction aggregation using least misery[1] is taken as baseline technique and is compared with different configurations for the proposed method.

GroupLens Research[a] in University of Minnesota provides a number of datasets for RS in the movies domain. In this experiment, ml-100k is used. For our purposes, it is needed to have information about groups of users, which MovieLens dataset does not provide. The group formation technique used selects random groups of a fixed number of members and the group sizes evaluated are 3, 5, 7 and 9. Hold-out validation scheme has been applied with 20%test, performing 100 independent executions. On each execution, 100 different groups were generated.

The proposed technique has four parameters:

(1) n: minimum members rating an item, $n = 1$ is used.
(2) MPD: ensure sufficient information for the GRS, $MPD = 80\%$ is used.
(3) μ: maximum disagreement value of ratings mantained, $\mu = \{1.0,\ 2.0,\ 3.0\}$ are tested.

[a]http://grouplens.org/

(4) KR: to decide what to do in special case 2: *true* to use an aggregated rating, *false* to remove all item ratings. Both cases tested.

The evaluation measure applied is MAE, to measure prediction error. The described experiment was executed in AMD Opteron 6272 with 16GB RAM. It took 8h 58m and its CPU process time was 7d 9h 18m.

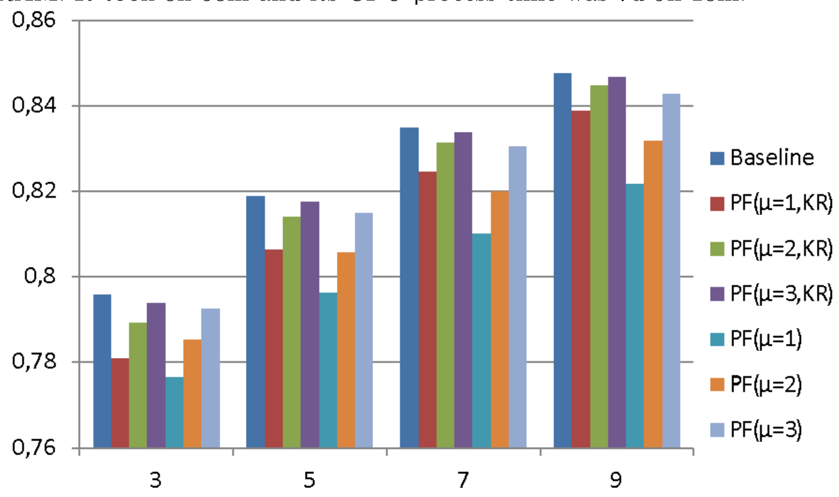

Fig. 4. Mean Average Error by group sizes.

In terms of algorithm performance, Fig. 4 shows the different configurations MAE by group size. As it shows, the proposed technique improves the baseline results for all parameter combinations.

Regarding the disagreement threshold, the smaller μ the better prediction (error decreases). This shows that as the filtering technique is applied in a stronger way (allows less disagreement, hence it deletes more ratings), the error decreases. Given that the experiment has been carried out with $\mu = \{1.0, 2.0, 3.0\}$ and the best prediction error is with $\mu = 1.0$, further experimentation is needed to determine if there is a value for μ between 0 and 1.0 that improves the results.

Looking at the results over different values for KR parameter, we can affirm that keeping a rating of movies on which there is no consensus is bad for the technique performance. Therefore, if an item is too controversial for the group, is better to remove all members ratings about it and perform group recommendation without it.

The proposed technique improves the baseline results for all group sizes, being $\mu = 1.0$ and $KR = false$ the best configuration for the tried group sizes, so filtering outlier ratings is a technique to consider in GRSs.

5. Conclusions

In this work, a filtering technique for group recommender systems is defined. This filtering process deletes users' ratings when members of the group do not agree on its preference. Therefore the proposed filtering process keeps ratings whose preferences agreed by the group and removes those ones with high disagreement. This technique has been compared with the baseline technique and shows improvements, which shows that disagreement ratings deletion improves recommendations results.

Acknowledgments

This work is partially supported by the Research Project TIN2012-31263, P10-AGR-6581, UJA2011/12/23 and ERDF funds.

References

1. M. O'Connor, D. Cosley, J. A. Konstan and J. Riedl, Polylens: a recommender system for groups of users, in *Proceedings ECSCW'01*, 2001.
2. F. Ricci, L. Rokach and B. Shapira, Introduction to recommender systems handbook, in *Recommender Systems Handbook*, (Springer, 2011) pp. 1–35.
3. J. Masthoff, Group recommender systems: Combining individual models, in *Recommender Systems Handbook*, (Springer, 2011) pp. 677–702.
4. J. F. McCarthy and T. D. Anagnost, Musicfx: an arbiter of group preferences for computer supported collaborative workouts, in *Proc. of CSCW '98*, (ACM, 1998).
5. L. Ardissono, A. Goy, G. Petrone, M. Segnan and P. Torasso, Intrigue: personalized recommendation of tourist attractions for desktop and hand held devices, in *Applied Artificial Intelligence, Vol 17 (8-9)*, (Taylor & Francis, 2003) pp. 687–714.
6. K. Myszkorowski and D. Zakrzewska, Using fuzzy logic for recommending groups in e-learning systems, in *Computational Collective Intelligence. Technologies and Applications*, (Springer, 2013) pp. 671–680.
7. N. Zheng and H. Bao, Flickr group recommendation based on user-generated tags and social relations via topic model, in *Advances in Neural Networks–ISNN 2013*, (Springer, 2013) pp. 514–523.
8. I. Cantador and P. Castells, Group recommender systems: New perspectives in the social web, in *Recommender Systems for the Social Web*, (Springer, 2012) pp. 139–157.
9. R. Meena and K. Bharadwaj, Group recommender system based on rank aggregation an evolutionary approach, in *Mining Intelligence and Knowledge Exploration*, (Springer International Publishing, 2013) pp. 663–676.
10. Y. Song, Z. Hu, H. Liu, Y. Shi and H. Tian, A novel group recommendation algorithm with collaborative filtering, in *International Conference on Social Computing (SocialCom) 2013*, Sept 2013.