Expert Systems With Applications 166 (2021) 114111

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

An entropy empowered hybridized aggregation technique for group recommender systems

Emre Yalcin [a], Firat Ismailoglu [a], Alper Bilge [b,*]

[a] Computer Engineering Department, Sivas Cumhuriyet University, 58140 Sivas, Turkey
[b] Computer Engineering Department, Akdeniz University, 07058 Antalya, Turkey

ARTICLE INFO

Keywords:
Group recommender system
Aggregation technique
Entropy
Additive utilitarian
Approval voting
Agreement without uncertainty

ABSTRACT

Group recommender systems aim to suggest appropriate products/services to a group of users rather than individuals. These recommendations rely solely on determining group preferences, which is accomplished by an aggregation technique that combines individuals' preferences. A plethora of aggregation techniques of various types have been developed so far. However, they consider only one particular aspect of the provided ratings in aggregating (e.g., counts, rankings, high averages), which imposes some limitations in capturing group members' propensities. Besides, maximizing the number of satisfied members with the recommended items is as significant as producing items tailored to the individual users. Therefore, the ratings' distribution is an essential element for aggregation techniques to discover items on which the majority of the members provided a consensus. This study proposes two novel aggregation techniques by hybridizing additive utilitarian and approval voting methods to feature popular items on which group members provided a consensus. Experiments conducted on three real-world benchmark datasets demonstrate that the proposed *hybridized* techniques significantly outperform all traditional methods. For the first time in the literature, we offer to use entropy to analyze rating distributions and detect items on which group members have reached no or little consensus. Equipping the proposed hybridized type aggregation techniques with the entropy calculation, we end up with an ultimate enhanced aggregation technique, *Agreement without Uncertainty*, which was proven to be even better than the *hybridized* techniques and outperform two recent state-of-the-art techniques.

1. Introduction

With the advent of internet-based technologies, people are encouraged to use online services to perform their daily activities, including shopping, watching movies, listening to music, and searching for news. Nevertheless, people are constantly facing a plethora of services/products, making it hard to find those that are likely to be of interest to them. This is also defined as an *information overload* problem (Adomavicius & Tuzhilin, 2005). Recommender systems aim to overcome this problem, recommending a set of relevant items while filtering out irrelevant information. Hence they ease the decision-making process of individuals and businesses (Bilge & Polat, 2013; Bobadilla et al., 2013; Masthoff, 2015).

In traditional settings, recommender systems produce referrals to individual users to satisfy individuals to the maximum extent reflecting their interests and tastes (Lu et al., 2015; Yera & Martínez, 2017). However, people prefer enjoying many activities with a group of people rather than alone, like watching movies with friends and dining at a restaurant for dinner with colleagues. Moreover, people have to act together with a crowd in some circumstances, like working out in a gym (McCarthy & Anagnost, 2000), traveling with a tour (Ardissono et al., 2003; McCarthy et al., 2006), and using mass transit. These cases require more complicated recommendation tools, as now the goal is not to satisfy individuals but the whole group with the produced recommendations.

Group Recommender Systems (GRSs) are developed as a response to the needs for providing a set of recommendations to a group of users (Ricci et al., 2011). They have found applications in several different domains such as movies (Crossen et al., 2002; O'Connor et al., 2001), restaurants (McCarthy, 2002), music (Crossen et al., 2002; Zhiwen et al., 2005), touristic attractions (Jameson, 2004; McCarthy et al., 2006), and TV programs (Goren-Bar & Glinansky, 2004). GRSs achieve this by considering the propensities of group members and their characteristics. These are obtained in two ways (Boratto et al., 2016): *(i)* combining group members' preferences or *(ii)* merging individual recommendations generated by a service provider. In both cases, various mathematical methods are utilized to aggregate

* Corresponding author.
E-mail addresses: eyalcin@cumhuriyet.edu.tr (E. Yalcin), fismailoglu@cumhuriyet.edu.tr (F. Ismailoglu), abilge@akdeniz.edu.tr (A. Bilge).

https://doi.org/10.1016/j.eswa.2020.114111
Received 9 April 2020; Received in revised form 12 September 2020; Accepted 8 October 2020
Available online 10 October 2020

preferences or individual recommendations, referred to as aggregation techniques (Seo et al., 2018). These techniques play a vital role in determining group preferences, thus directly affect the quality of produced group recommendations.

GRSs commonly produce group recommendations for a set of users that has similar tastes through group recommendation methods rather than providing individual personalized recommendations because of cost or context. However, most of the time, groups of people that share interests are unknown in advance. Therefore, often it is a must to detect the groups before producing recommendations automatically, which raises the issue of *automatically identified groups* (Boratto & Carta, 2014; Boratto et al., 2016). To detect such groups, one of the well-known clustering methods, such as k-means (Boratto & Carta, 2014; Boratto et al., 2016), or hierarchical clustering (Cantador & Castells, 2011), is usually employed. The groups constructed in this manner are very likely to be coarse-grained, as no clustering algorithm can yield perfect clusters. As a result, the total number of users pleased with a GRS recommendation will be less; one of the most unwanted scenarios.

In imperfectly constructed groups, the aggregation techniques that unify group members' preferences might fall short of satisfying members. Such a drawback occurs mainly when a single aggregation technique is employed, as it reflects only one particular aspect of provided ratings in aggregating. More clearly, aggregation techniques come in several flavors based on varying aspects of provided ratings as some rely on counting frequencies (Crossen et al., 2002; Lieberman et al., 1999), others on their ranking (Álvarez Márquez & Ziegler, 2016; Boratto et al., 2016; Masthoff, 2015), or high-averages (Jameson, 2004; McCarthy et al., 2006), and so on Seo et al. (2018). When the groups inherited in the aggregation phase were not appropriately constructed, we deal with imbalanced ratings, which are challenging to be aggregated by a single technique. When this is the case, one needs to employ multiple aggregation techniques at once, each handling one particular dimension.

Another problem with having heterogeneous groups in a GRS is that there may be items where group members reached no consensus. For example, suppose the distinct values of the ratings that an item received within a group have similar frequencies (i.e., they are evenly distributed). In that case, it is hard to claim that all of the group members agree with that item. It is essential to reduce the chances of such items to be included in the recommendation list since maximizing the number of group members satisfied with recommendation outputs is as significant as providing perfectly suitable predictions for individuals. To this end, it is necessary to detect such items, which can be achieved by analyzing their ratings' distributions. In analyzing rating distribution, the standard deviation is usually considered (Seo et al., 2018). If the deviation is high, it is interpreted that the members have diverse opinions. Although this seems a reasonable approach, using the standard deviation to analyze the dispersion of ratings while combining individual preferences raises some fundamental problems, as elaborated below.

The standard deviation measures the dispersion of variables of continuous type, by its nature. However, user preferences expressed by ratings are of discrete type, with little exception. This inconsistency causes more problems when the ratings vary in a narrow range, such as [0, 1] or [1, 5]. Even in a ten-star rating system, the standard deviation may not capture the overall behavior of the ratings' dispersion. Because it is likely to observe multiple peaks, yielding a multi-modal distribution, as the ratings are allowed to have more distinct values, wherein the standard deviation fails to explain the dispersion. This problem is more apparent in the presence of large or imperfect groups, as it becomes more likely to have subgroups with differing tastes.

In order to cope with the problems mentioned above, we focus on developing novel aggregation techniques to improve group recommendations' accuracy and fairness and increase overall group satisfaction in the present study. The proposed aggregation techniques are obtained by hybridizing traditional methods of assembling group preferences and filtering out debatable items based on their rating distribution's uncertainty. The following summarizes the main contributions of the present study.

1. We scrutinize the baseline traditional aggregation techniques commonly utilized in group recommendation studies to identify their limitations in aggregating individual preferences. We also present a comprehensive classification of existing GRSs according to various properties such as utilized aggregation technique, application domain, type of acquisition of user ratings, and so on.
2. We introduce two novel aggregation techniques termed as *hybridized* techniques that suitably combines additive utilitarian and approval voting methods in two different ways to provide more accurate and satisfying group recommendations.
3. We further propose an enhanced ultimate aggregation technique, termed *agreement without uncertainty (AwU)*, that is built on top of *the hybridized* techniques. Specifically, the AwU robustly considers the distribution of group members' ratings by utilizing the information entropy and consequently produces group recommendations that ensures the maximum number of satisfied individuals.

We organize the rest of the study as follows: The next section explains the traditional aggregation techniques used in GRSs and analyzes them according to their combining strategies in detail. Section 3 presents a detailed literature summary of well-known GRSs. Section 4 introduces the proposed group recommendation scheme, including novel aggregation techniques, and the following section demonstrates experimental work, achieved results, and gained insights. Finally, Section 6 concludes the study and gives directions for future research.

2. Aggregation techniques

As mentioned in the previous section, GRSs attempt to aggregate group members' preferences or predictions to end up with group preferences. The preferred aggregation technique accomplishes this task. In the literature, a wide range of aggregation techniques have been proposed so far to address different needs in different scenarios. The most prominent ones include average (Ardissono et al., 2003; Christensen & Schiaffino, 2011; Jameson, 2004; Liu et al., 2016; McCarthy et al., 2006; Quijano-Sanchez et al., 2011; Zhiwen et al., 2005), average without misery (Chao et al., 2005), additive utilitarian (Agarwal et al., 2017; Boratto et al., 2016; Kaššák et al., 2016; McCarthy, 2002; Yalcin et al., 2019), multiplicative (Christensen & Schiaffino, 2011), approval voting (Boratto et al., 2016; Lieberman et al., 1999; Seo et al., 2018), simple count (Crossen et al., 2002), plurality voting (Salehi-Abari & Boutilier, 2015), most pleasure (Ahmad et al., 2017; Boratto et al., 2016), least misery (Agarwal et al., 2017; Boratto et al., 2016; Christensen & Schiaffino, 2011; O'Connor et al., 2001), Borda count (Álvarez Márquez & Ziegler, 2016; Boratto et al., 2016), Copeland rule (Masthoff, 2015; Yalcin et al., 2019), most respected person (Masthoff, 2015), and upward leveling (Seo et al., 2018).

In the following, we attempt to categorize the aggregation techniques used in GRSs into seven kinds based on how to combine given or predicted ratings. We also provide a user–item matrix shown in Table 1 to exemplify the aggregation techniques to be categorized. Concretely, this matrix represents four users that construct a group and six items that they rate on a five-star scale. Here $\perp$ denotes the unrated items by the users.

Providing consensus: A common practice in GRSs is to incorporate all group members in estimating group rating. In doing so, a consensus is provided among the group members. Thus, the aggregating techniques following this practice are termed as *providing consensus*. They use fundamental arithmetic operations such as

Table 1
A user–item matrix to exemplify aggregation techniques.

	i_1	i_2	i_3	i_4	i_5	i_6
u_1	3	2	$\perp$	1	$\perp$	1
u_2	4	$\perp$	$\perp$	2	5	3
u_3	$\perp$	4	3	4	$\perp$	$\perp$
u_4	1	$\perp$	3	$\perp$	3	2

averaging, addition, and multiplication and well-known examples of this kind are *Average* (Avg) (Ardissono et al., 2003; Christensen & Schiaffino, 2011; Jameson, 2004; Liu et al., 2016; McCarthy et al., 2006; Quijano-Sanchez et al., 2011; Zhiwen et al., 2005), *Average without Misery* (AwM) (Chao et al., 2005), *Additive Utilitarian* (AU) (Agarwal et al., 2017; Boratto et al., 2016; Kaššák et al., 2016; McCarthy, 2002; Yalcin et al., 2019), and *Multiplicative* (Mul) (Christensen & Schiaffino, 2011). Concretely, Avg determines group ratings by simply calculating the averages of individual ratings. AwM is a variant of Avg in the sense that it disregards the ratings below a user-defined threshold in calculating the average. So, AwM ignores an item if it receives a rating below the threshold. On the other hand, AU sums up individual ratings, while Mul multiplies them to have group ratings. According to the ratings given in Table 1, these techniques determine group ratings as in Table 2, where the threshold value for AwM is selected as 3.

The techniques *providing consensus* are easy to implement and effective. However, they do not always assure revealing the actual taste of a group. In particular, if an item is rated high by just a few users in a group, which is quite likely in GRSs, then those members who did not rate that item (vast majority of the group) may not favor it. The same holds for AU and Mul, too. In case an item is rated by most of the members of a group by a low rating, AU and Mul scores such items high, which could eventually lead the GRS recommending the item. Moreover, in the presence of an item rated by quite many group members, then group rating estimated by Mul for the corresponding item converges to infinity, which consequently entails the overflow problem.

Counting frequency of ratings: In recommender systems, it is usually assumed that users provide ratings for items they are interested in. Based on this assumption, counting the number of user ratings for an item can be utilized to quantify the group interest on that item, which is an indicator of how much popular an item is within the group. In general, the aggregation techniques identifying the most popular items in a group are termed as techniques *counting frequency of ratings*. *Simple Computation* (SC) (Crossen et al., 2002) and *Approval Voting* (AV) (Boratto et al., 2016; Lieberman et al., 1999; Seo et al., 2018) are the prominent examples of this kind. Specifically, SC counts the number of times that an item is rated by the group members, whereas AV ignores ratings below a predefined threshold (i.e., negative ratings) while counting. Table 3 provides an example of these techniques, where the threshold value for AV is selected as 3.

SC's main drawback is that it considers an item recommendable, even if it receives negative ratings from the majority of a group. AV overcomes this problem taking only ratings above a threshold (i.e., positive ratings) into account. Nevertheless, excluding all negative ratings makes it challenging to understand the actual opinion on an item.

Highest/lowest rating The aggregation techniques in this category consider the extreme ratings as vital. Such extreme ratings can be considered either the highest or the lowest rating, where the corresponding aggregation techniques are generally referred to as *Most Pleasure* (MP) (Ahmad et al., 2017; Boratto et al., 2016) and *Least Misery* (LM) (Agarwal et al., 2017; Boratto et al., 2016; Christensen & Schiaffino, 2011; O'Connor et al., 2001), respectively.

MP picks the highest rating among the members' ratings for an item, whereas LM picks the lowest rating. When a group consists of a large number of members, MP (resp. LM) tends to generate the same high (resp. low) rating for almost every item; as it becomes likely that for each item, at least one member in the group gives a high (resp. low) rating. In this case, they both fail to differentiate items that are preferred by the group. Table 4 presents group ratings determined by the MP and LM techniques.

As MP, *Plurality Voting* (PV) (Salehi-Abari & Boutilier, 2015) also considers that the highest ratings are essential for aggregation purposes. However, PV differs from MP in that it gets the highest rating for each user as opposed to MP getting the highest rating for each item. Besides, PV generates a recommendation list of items with the highest ratings, whereas the techniques mentioned above aggregate the group members' aggregate preferences.

First, PV takes the item with the highest rating from each member, then puts the item received the highest rating by the majority of the group at the top of the recommendation list. This item is then removed from the favorite sets of the group members. Again, it takes the item with the highest rating from each member regarding the remaining items, then adds it to the second-best recommendation list. This process repeats until the recommendation list is completed. To grasp how PV works, see Table 5 produced from Table 1.

Ranking priority: The aggregation techniques in *ranking priority* category rely on sorting the ratings as in PV, but differ in that they provide a rank regarding their position in the sorted list (Masthoff, 2015). *Borda Count* (BC) (Álvarez Márquez & Ziegler, 2016; Boratto et al., 2016) is the popular example of this category. In BC, the following is performed for each member: items get a ranking in ascending order where the item with the lowest rating gets the rank of 0. In case of ties, average ranks of corresponding items are assigned. Once this has been accomplished, BC sums up the rankings that each item received. See Table 6 constructed using Table 1.

Both PV and BC are criticized for sorting involved, mainly because sorting the ratings becomes infeasible in some circumstances. For example, given ratings might manifest a uniform distribution for several users in a group. Worse, the ratings are on a five-star scale in general (meaning that there are five distinct ratings in total), whereas, in a typical recommendation system, there are at least hundreds of items. Thus, it is inevitable that so many items share the same rating following the pigeonhole principle. Furthermore, the sorting process entails high-computation time.

Comparing ratings: The most preferred items of a group can also be determined via the relative importance of the items. This task can be performed by considering the items' mutual preference status according to the group members' ratings; hence, the aggregation techniques in this category are termed as *comparing ratings*. *Copeland Rule* (CR) (Masthoff, 2015; Yalcin et al., 2019) is the most salient example of *comparing ratings*.

For example, according to the ratings given in Table 1, CR aggregates individual ratings per item, as shown in Table 7. Here, the kth row and lth column ($k, l \in \{1, \ldots, 6\}, k \neq l$) of the table shows how many times the item k was preferred over the item l among the group members. For example, the first item (i_1) was preferred over the second item (i_2) three times, whereas the second item was preferred over the first item only

Table 2
Group ratings by techniques *providing consensus*.

	i_1	i_2	i_3	i_4	i_5	i_6
Avg	2.67	3	3	2.33	4	2
AwM	–	–	3	–	4	–
AU	8	6	6	7	8	6
Mul	12	8	9	8	15	6

once. Thus corresponding scores are calculated as $3 - 1 = +2$ and $1 - 3 = -2$ respectively. For each item, the final CR score is obtained by summing each pairwise comparison that the item involved. However, the main shortcoming of CR is that it has to compare all pairs of items, which incurs a high computational cost, especially when the number of available items proliferates.

Based on the influential member: Some members of a group can influence other individuals' decision-making process, and they are known as the influential members of the group. Preferences of an influential member in a group can be taken as the group choice. The aggregation techniques that rely on this idea are termed as techniques *based on the influential member*. *Most Respected Person* (MRP) (Masthoff, 2015), the well-known example of this category, takes the ratings of the most influential member in a group to construct the group profile. It is needless to say that relying on a single user's opinions while disregarding the others in the group is often not the ideal aggregation technique, in particular, having large groups. Above all, it is also unclear who should be selected as the most respected person in a given group. To exemplify MRP, we provide Table 8, where the most respected person is selected as u_2.

Considering rating distribution: The principal motivation of GRSs is to satisfy as many group members as possible with the recommended items, which requires to have perfect groups, i.e., those consisting of all like-minded users. However, it is usually not the case. Nevertheless, even for imperfect groups, some items on which the vast majority of the group agree. In order to place these items into the recommendation list, the employed aggregation technique should consider the distribution of ratings within the group. In the literature, these aggregation techniques are generally referred to as *considering rating distribution* (Seo et al., 2018).

Upward Leveling (UL) (Seo et al., 2018) has been recently introduced as an enhanced aggregation technique that considers the distribution of the ratings per item. To this end, UL calculates the standard deviation (SD) of the ratings of an item; then combines it with Avg and AV scores of the item to arrive at the ultimate aggregation. This combination is performed by calculating the weighted average, where the weights are randomly selected from the set: $\{0, 0.1, \ldots, 0.9, 1\}$. UL initially transforms original ratings into [0, 1] scale and applies a min–max normalization process for group scores calculated by AV. We provide an example in Table 9 to show how UL works in practice. Note that the weights assigned to Avg, AV, and SD values are the same: $1/3$ in the example.

Even though UL is the first of its kind to consider the distribution of the ratings while aggregating, it only works well when these distributions have only one peak (i.e., uni-modal distribution), since UL employs standard deviation. However, it is quite likely to come across more than one peak in a rating distribution (i.e., multi-modal distribution), which occurs especially when the constructed groups are large/imperfect, and the rating scale is wide (e.g., 10-star scale).

Table 3
Group ratings by techniques *counting frequency of ratings*.

	i_1	i_2	i_3	i_4	i_5	i_6
SC	3	2	2	3	2	3
AV	1	1	0	1	1	0

Table 4
Group ratings by MP and LM techniques.

	i_1	i_2	i_3	i_4	i_5	i_6
MP	4	4	3	4	5	3
LM	1	2	3	1	3	1

Table 5
Recommendation list produced by PV technique.

	Recommendation list					
	1	*2*	*3*	*4*	*5*	*6*
u_1	i_1	i_1	i_2	i_4, i_6	i_6	–
u_2	i_5	i_1	i_6	i_6	i_6	–
u_3	i_2, i_4	i_2, i_4	i_2, i_4	i_4	i_3	i_3
u_4	i_3, i_5	i_3	i_3	i_3	i_3	i_3
PV	i_5	i_1	i_2	i_4	i_6	i_3

Table 6
Group ratings by BC technique.

	i_1	i_2	i_3	i_4	i_5	i_6
u_1	3	2	$\perp$	0.5	$\perp$	0.5
u_2	2	$\perp$	$\perp$	0	3	1
u_3	$\perp$	1.5	0	1.5	$\perp$	$\perp$
u_4	0	$\perp$	2.5	$\perp$	2.5	1
BC	5	3.5	2.5	2	5.5	2.5

Table 7
Group ratings by CR technique.

	i_1	i_2	i_3	i_4	i_5	i_6
i_1	■	−2	0	0	+1	−1
i_2	+2	■	−1	0	0	0
i_3	0	+1	■	+2	0	0
i_4	0	0	−2	■	0	+1
i_5	−1	0	0	0	■	−1
i_6	+1	0	0	−1	+1	■
CR	+2	−1	−3	+1	+2	−1

Table 8
Group ratings by MRP technique.

	i_1	i_2	i_3	i_4	i_5	i_6
MRP	4	$\perp$	$\perp$	2	5	3

Table 9
Group ratings by UL technique.

	i_1	i_2	i_3	i_4	i_5	i_6
Avg	0.53	0.6	0.6	0.46	0.8	0.4
AV	1	1	0	1	1	0
1-SD	0.75	0.8	1	0	0.75	0.84
UL	0.76	0.8	0.53	0.48	0.85	0.41

3. Related work

Since the last two decades, several GRSs have been developed for different scenarios in various domains: music (Chao et al., 2005; Christensen & Schiaffino, 2011; Crossen et al., 2002; McCarthy & Anagnost, 2000; Zhiwen et al., 2005), movies (Boratto et al., 2016; Christensen & Schiaffino, 2011; Mahyar et al., 2017; Ntoutsi et al., 2012; O'Connor et al., 2001; Quijano-Sanchez et al., 2011), points-of-interest (restaurants (McCarthy, 2002), touristic attractions (Álvarez Márquez &

Ziegler, 2016; Ardissono et al., 2003; Basu Roy et al., 2014; Jameson, 2004; McCarthy et al., 2006), etc.), browsing (Lieberman et al., 1999), books (Ahmad et al., 2017), and TV programs (Goren-Bar & Glinansky, 2004) to name but a few. The aggregation techniques involved in these applications vary widely, as no single aggregation technique can achieve high performance in all applications.

MusicFX (McCarthy & Anagnost, 2000) is an intelligent environment that selects a radio station for workout music for a group of users at a gym by estimating probabilities of music genres being favorite to the users. MusicFX employs AwM as the aggregation technique, thus only considering genres where users' preferences are all above a threshold. Speaking of GRSs in the domain of music, Flytrap (Crossen et al., 2002) is another well-known example. Flytrap creates a virtual DJ that composes a playlist for people in a particular room. In doing so, the virtual DJ first reveals the people's musical tastes; and then blends this with background knowledge of interrelation between music genres, the transition between songs, and the influence between artists. The aggregation technique opted for Flytrap is a variant of SC. Other salient examples of GRSs in the music domain are Adaptive Radio (Chao et al., 2005) and the Adaptive In-Vehicle Multimedia System (Zhiwen et al., 2005). More specifically, Adaptive Radio is a server that selects music to be played for a group of users, relying on the information about what kind of music they do not want to hear, as this type of information is more comfortable to gather. Adaptive Radio aggregates such negative preferences by a without misery aspect. The Adaptive In-Vehicle Multimedia System, on the other hand, recommends a series of songs for a group of passengers traveling in a vehicle together, as its name suggests. This system merges the selected features of passengers by utilizing the Avg technique to construct group profiles.

PolyLens (O'Connor et al., 2001) is an extension of the famous MovieLens and provides movie recommendations for groups of users instead of individuals, which is the case in MovieLens. As an aggregation technique, PolyLens utilizes the LM in combining user preferences for the movies. Additionally, in PolyLens, a novel questionnaire is designed to investigate users' satisfaction from the system. The jMusicGroupRecommender and jMovieGroupRecommender (Christensen & Schiaffino, 2011) are the entertainment GRSs providing group recommendations for music and movie, respectively. These systems use several group recommendation approaches in a harmony in which Mu, Avg, and LM are utilized as aggregation techniques. Another example in the movie domain is the gRecs (Ntoutsi et al., 2012), which uses the LM and MR techniques. The gRecs first creates clusters of similar interests and then produces top-N movie recommendations by following a collaborative strategy. HappyMovie (Quijano-Sanchez et al., 2011) is a Facebook application recommending movies to groups of people based on the interests of group members and the trust among them. The HappyMovie employs Avg and LM for combining individual preferences. IBGR (Barzegar Nozari & Koohi, 2020) is an enhanced approach for recommending movies to groups by considering social relationships among the group members during the aggregation process. Specifically, this method determines group members' influence on each other by calculating the similarity and the trust among the users. It then utilizes them in weighting the individuals' preferences via the Avg technique. TruGRC (Wang et al., 2019) also relies on social relationships based on trust among group members, where the calculated trust is incorporated in the phase of aggregation governed by the Avg as in IBGR. Finally, The NNMG (Castro et al., 2018) has been introduced as a GRS that detects the natural noise present in individuals' preferences. The NNMG removes the noise to eliminate its effects in the movie recommendations produced for a group of users. Also, the NNMG utilizes both Avg and LM techniques to aggregate both individual preferences and provided recommendations.

Pocket Restaurant Finder (McCarthy, 2002) produces a list of recommended restaurants for a group of people. This system utilizes the AU technique to combine their preferences on food (e.g., taste, price category, restaurant amenities, the cuisine type) and location. INTRIGUE (Ardissono et al., 2003), Travel Decision Forum (Jameson, 2004), and CATS (McCarthy et al., 2006) are some other well-known GRS applications in the context of tourism activities. More specifically, INTRIGUE recommends a sequence of tourist attractions for guided tours based on the tour participants' characteristics in favor of children and the disabled. On the other hand, Travel Decision Forum produces a single recommendation on where to go for a group of people planning to take a vacation together, helping their decision-making process. Finally, CATS recommends ski-packages that best suit a group's demands, and all group members' needs. To construct group profiles, CATS uses the Avg technique to combine the critiques of the members. Thinking of visiting web pages as visiting actual psychical places, Let's Browse (Lieberman et al., 1999) that recommends which page to visit next to a group can be given here. Let's Browse assists user groups in web-browsing by utilizing matching scores determined by comparing user-profiles and web pages. This system considers web pages with the matching score above a threshold only, as in the AV technique. Finally, GIST (Ji et al., 2018) is a social-media GRS providing group recommendations by constructing group profiles based on a probabilistic case-based model. Compared to the traditional aggregation techniques, this system considers individual interests and common appeals of the subgroups.

Kaššák et al. (2016) introduce a group recommendation algorithm of hybrid type by merging items recommended by a content-based algorithm and those recommended by a collaborative filtering algorithm to get the final suggested items for a group. Here, the goal is to recommend very few items to a group (very top-N recommendations). The CoGrec system (Liu et al., 2016) extracts user profiles through non-negative matrix factorization, and then uses these profiles to detect groups as overlapping communities. When it comes to aggregating user preferences, the system averages overlapping community memberships of the users. Mahyar et al. (2017) propose using the concept of centrality in graph theory to estimate how influential the members in a group, which enables user centrality measures to be used to weight ratings of group members while aggregating. Recently, Seo et al. (2018) have proposed an enhanced aggregation technique named Upward Leveling, which considers the deviation in the user preferences as a vital element for group recommendation. This deviation is exploited in aggregating user ratings; indeed, combining AU and AV scores altogether, where the blending is achieved through a weighted average. However, the weights are determined manually by hand-tuning.

There exist two remarkable experimental studies in the literature comparing the aggregation techniques. Boratto et al. (2016) analyze the performance of some benchmark aggregation techniques, including AU, AV, BC, LM, and MP, in varying group sizes. Yalcin et al. (2019) consider a more extensive set of aggregation techniques and comprehensively examine their effectiveness. These studies show that the sizes of the groups in a GRS directly impact the aggregation technique's performance. The larger the groups are, the less satisfied the users with the items recommended by the GRS in general. Also, increasing the size of the recommendation list may impair the performance as well. Finally, it has been experimentally shown that there exists no one-size-fits-all aggregation technique that can identify group preferences in all application domains.

When a group of people has to make a decision, they usually interact with each other negotiating within the group to achieve a consensus. There exist some methods in GRSs that consider this phenomenon in recommending items to the group (Bedi et al., 2014; Bekkerman et al., 2006; Nguyen & Ricci, 2018; Villavicencio et al., 2019). Expressly, these methods represent each user in a group with an agent, and then let them negotiate on behalf of the users. A set of items on which most of the agents agree emerges a result of the cooperative negotiation. The major advantage of this approach over the remaining GRSs is to allow for users' tolerance and users' interaction in determining an item for the group, which may allow for satisfaction of the group as a whole. More specifically, a non-negative utility function is defined for each user in a

group that maps proposed items to its satisfaction value. Also, a utility function is defined for the group to evaluate the utility of recommended items to the group as a whole. Here, we note in passing that a group utility function can be a known aggregation technique such as AU and Avg. Having defined the individual and the group utility functions, a protocol specifying the range of legal moves for each agent at each step of the negotiation is determined.

In fact, the negotiation-based GRSs differ mainly in the protocol they use. For example, Bekkerman et al. (2006) employs a simple protocol called "AlternatingOffers" that only allows for placing an offer and accepting one of the previously placed offers. On the other hand, Villavicencio et al. (2019) employs a somewhat complicated protocol called "Monotonic Concession Protocol" that governs a broader range of moves, such as who makes the next concession, how much an agent can concede, and so on. Nevertheless, in the presence of non-homogeneous groups, i.e., those consisting of subgroups, the negotiation may fail to reach an agreement due to the conflicting preferences of the users, which in turn leads to the negotiation-based GRSs producing no recommended items to the group (Bekkerman et al., 2006; Villavicencio et al., 2019).

We summarize the well-known GRSs in Table 10 according to the following seven categories: system name, utilized aggregation technique, application domain, type of acquisition of user ratings, group size considered in the experiments, aggregation strategy used in estimating group ratings (combining individual preferences or produced recommendations), and finally group type utilized in the system (being real or virtual).

As is seen from the present section, various GRSs have been developed so far for different purposes. The vast majority of them (Ardissono et al., 2003; Chao et al., 2005; Crossen et al., 2002; McCarthy et al., 2006; O'Connor et al., 2001; Zhiwen et al., 2005), however, usually use a single aggregation technique only, which leads to construct group profiles from a single point of view. Therefore, there is a need for employing multiple aggregation techniques in harmony to capture different aspects of group profiles. Besides, the distribution of user preferences within a group plays a crucial role in analyzing group members' propensities appropriately. The off-the-shelf aggregation techniques addressing such distribution employ standard deviation (Seo et al., 2018), which comes into play only when the distribution is uni-modal. However, they fail to identify preferable items when the distribution of the user ratings is multi-modal, which is the case, especially in the presence of large user groups nor large-scale rating systems.

4. A group recommendation scheme based on novel aggregation techniques

This section presents our group recommendation scheme that provides high-quality group referrals and consists of two main steps, as depicted in Fig. 1. Firstly, we explain how groups of users utilized in the proposed group recommendation scheme are identified. Then, we introduce novel aggregation techniques, two variants of *hybridized* and *agreement without uncertainty (AwU)*, for predicting group ratings to be used in providing top-*N* group recommendations. More specifically, the *hybridized* techniques rely on two different combinations of additive utilitarian and approval voting methods. The AwU, on the other hand, is an enhanced aggregation technique built on top of the *hybridized* techniques and utilizes information entropy to analyze the distribution of group members' ratings.

4.1. Identification of user groups

Although a few studies use established groups in GRS literature, it is typical that groups of users with similar interests are not predefined. Therefore, most of the existing GRS's initial procedure is to partition users into groups automatically, which is referred to as *automatic identification of groups* (Boratto et al., 2016). However, the performance of utilized GRSs is strongly correlated with identifying groups having similar users since it is easier to satisfy like-minded users rather than a randomly ensembled mass. Also, the automatic identification of user groups is beneficial because of two reasons: *(i)* grouping users is a continuous process requiring regular updates due to the changes in the interests of users over time, and *(ii)* manually partitioning users into groups becomes challenging with the increasing number of users.

To detect groups of users by considering the preferences of individuals in a community, most of the existing studies utilize one of the two common methods in the following: *(i)* identifying groups of users by computing correlations among all individuals (Baltrunas et al., 2010), and *(ii)* utilizing either a traditional clustering algorithm (*k*-means (Boratto et al., 2016), *k*-medoids (Khazaei & Alimohammadi, 2018), etc.) or hierarchical clustering techniques (Cantador & Castells, 2011). Although the former method successfully detects groups of similar users, the computation time required to calculate correlations among users dramatically increases as the number of users/items in the system proliferates, which causes the problem of time complexity. On the other hand, the latter method is a suitable way of identifying groups of like-minded people and is comparatively more efficient in terms of time complexity. Therefore, to construct user groups, we follow the latter method by using the *k*-means algorithm as it is a simple and efficient clustering method and can be applied to almost all data types (Boratto & Carta, 2014; Boratto et al., 2016).

We detect groups of users with similar tastes by simply applying the *k*-means algorithm on the original user–item rating matrix. Having user groups are determined, the predicted ratings for each group are calculated by the aggregation techniques described in detail in the following sections.

4.2. The hybridized techniques

The aggregation techniques aim to construct group profiles that embody the tastes of a bunch of individuals as a whole (Boratto et al., 2016). Group recommendations are usually produced based on such group profiles, making the selection of the aggregation technique critical. Although the literature is rich in aggregation techniques of various types, they follow a unique strategy to combine individuals. Each of them comes with some limitations, as previously discussed in Section 2. In other words, each technique builds a group profile that consists of the predicted group ratings for items by aggregating preferences of members from a different point of view. To determine the preferable items within a group and consequently improve overall satisfaction, it is wise to employ more than one aggregation technique together in harmony, which is equivalent to develop a high-quality *hybridized* aggregation technique.

In attempting to hybridize the aggregation techniques, the first arising question is which techniques should be selected as base approaches in the hybrid model. One who faces this problem should take into account their expectations from the aggregation technique to be constructed. Our expectations are centered on providing consensus among group members and featuring items that are the favorite of most members. To meet the first expectation, we propose to employ the Additive Utilitarian (AU), which determines a group's preference on an item by summing up ratings it received by all of the members. AU, thereby, allows every member with an opinion on an item in question to contribute the group's decision for that item. Also, to address the second expectation, we propose to employ the Approval Voting (AV), which counts the number of ratings above a given threshold. Therefore it allows the final aggregation technique to be biased towards highly popular items among the group members.

In Section 2, we discuss the shortcomings of the aggregation techniques. Specifically, AU carries the risk of selecting an item with a low vote from most group members. The sum of these ratings may be significant, mostly when several members have rated the item. On the other hand, AV ignores items with low ratings in favor of highly-rated

Table 10
Classification of existing GRSs.

Group recommender system	Aggregation technique	Domain	User preferences	Group size	Aggregation type	Group type
Let's Browse (1999) (Lieberman et al., 1999)	AV	Browsing	E	n/a	P	R
MusicFX (2000) (McCarthy & Anagnost, 2000)	AwM	Music	I	M	P	R
PolyLens (2001) (O'Connor et al., 2001)	LM	Movie	E, I	S	P	R
FlyTrap (2002) (Crossen et al., 2002)	SC	Music	I	S	P	V
Pocket RestaurantFinder (2002) (McCarthy, 2002)	AU	Restaurant	E	S	P	R
INTRIGUE (2002) (Ardissono et al., 2003)	Avg	Travel	E, I	S	P	V
Travel Decision Forum (2004) (Jameson, 2004)	Avg	Travel	E	S	P	R
FIT-Family (2004) (Goren-Bar & Glinansky, 2004)	AU	TV	E, I	S	P	V
Adaptive Radio (2005) (Chao et al., 2005)	AwM	Music	E	S	P	R
CATS (2006) (McCarthy et al., 2006)	Avg	Travel	E, I	S	R	R
In-vehicle multimedia recommender (2006) (Zhiwen et al., 2005)	Avg	Music	E	S	P	R
jMusicRecommender & jMovieRecommender(2006) (Christensen & Schiaffino, 2011)	Mul, LM, Avg	Movie, Music	NF, I	M	P, R	V
Happy Movie (2011) (Quijano-Sanchez et al., 2011)	Avg, LM	Movie	I	S	R	R
gRecs (2012) (Ntoutsi et al., 2012)	LM, MR	Movie	E	n/a	R	V
FlexiFeed (2014) (Basu Roy et al., 2014)	LM, Avg	Movie, Travel	NF, I	L	P	V
POSN (2015) (Salehi-Abari & Boutilier, 2015)	PLU, BC	Movie	E, I	M	R	V
Boratto et al. (2016) (Boratto et al., 2016)	AU, AV, BC, LM, MP	Movie	E	L	P, R	V
Hootle+ (2016) (Álvarez Márquez & Ziegler, 2016)	BC	Travel	E	S	P	R
CoGrec (2016) (Liu et al., 2016)	Avg	Movie	E	L	P	V
Kassak et al. (2016) (Kaššák et al., 2016)	AU	Movie	E	S	R	V
Ahmad et al. (2017) (Ahmad et al., 2017)	LM, Avg, MP	Book	E, I	S	P	V
Agarwal et al. (2017) (Agarwal et al., 2017)	AU, LM	Movie	E	L	R	V
Mahyar et al. (2017) (Mahyar et al., 2017)	Weighted(Avg)	Movie	E	M	P	V
Seo et al. (2018) (Seo et al., 2018)	UL	Movie	E	L	P	V
GIST (2018) (Ji et al., 2018)	n/a	Social	I	S, M	P	V
NNMG (2018) (Castro et al., 2018)	Avg, LM	Movie	E	S, M	P, R	V
MAGRes (2019) (Villavicencio et al., 2019)	n/a	Movie, Travel	E	S	P	V
TruGRC (2019) (Wang et al., 2019)	Avg	Social	E	S, M	P	V
IBGR (2020) (Barzegar Nozari & Koohi, 2020)	Avg	Movie	E	S, M	P	V

User preferences = *Implicit (I), Explicit (E), Negative feedback (NF)*
Group size = *Small (S): # members in group < 10, Medium (M): 10 ≤ # members in group < 100, Large (L): 100 ≤ # members in group*
Aggregation type = *Preferences (P), Recommendations (R)*
Group type = *Real (R), Virtual (V)*

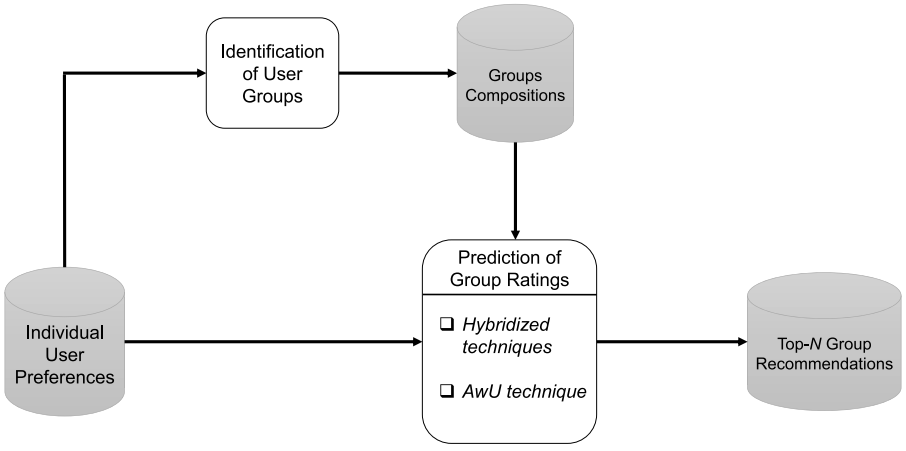

Fig. 1. Group recommendation scheme based on novel aggregation techniques.

items. However, AV evaluates items considering only the preferences of a small subgroup, which leads to diminishing overall satisfaction. As a result, we propose to employ AU and AV together, aiming to create synergy in eliminating their shortcomings.

We propose two novel aggregation techniques of the hybridized type where AU and AV are the base techniques. The proposed techniques determine group ratings by taking into account *(i)* the sum of preferences of members in the group (achieved by employing AU) *(ii)* the number of evaluations that are highly rated (achieved by employing AV). This way, the proposed techniques enable featuring popular items where a consensus is somehow reached.

The developed *hybridized* aggregation techniques differ in which base technique plays a more decisive role in group ratings. We shall denote these techniques as AU_{AV} and AV_{AU}, where the driving force is

AU and AV respectively. The following explains how group ratings are computed using AU_{AV} and AV_{AU}.

- AU_{AV}: This *hybridized* technique utilizes AU as the driving force, while AV is an additive factor on the final group ratings, as formulated in Eq. (1). Here $R_{g,i}^{AU_{AV}}$ denotes the final rating of a group g for an item i, where $AU_{g,i}$ and $AV_{g,i}$ denotes the ratings of g for i calculated by using the techniques AU and AV, respectively.

$$R_{g,i}^{AU_{AV}} = AU_{g,i} + (AU_{g,i} \times AV_{g,i}) \tag{1}$$

- AV_{AU}: The second technique, on the other hand, values AV as the decisive factor on final group ratings, while it employs AU as an auxiliary influencer. $R_{g,i}^{AV_{AU}}$ denotes the final rating of group g for item i calculated by AV_{AU}. This calculation is formulated in Eq. (2).

$$R_{g,i}^{AV_{AU}} = AV_{g,i} + (AV_{g,i} \times AU_{g,i}) \tag{2}$$

Note that, in AU, the maximum value corresponds to the multiplication of the highest vote in the rating scale with the number of members in the group (occurs when all members provide the highest vote for each item). On the other hand, in AV, the maximum value of a group rating determined by AV is equal to the number of members in the group (i.e., if all members provide a rating above the threshold for each item). Hence, AU's values vary in a wider range than that of AV, which requires a normalization process to make the values comparable before calculating final group ratings by the proposed *hybridized* techniques. For this purpose, we first transform group ratings obtained by AU and AV into $[0, 1]$ scale through min–max normalization, and then combine them using either AU_{AV} or AV_{AU}.

4.3. The agreement without uncertainty technique

A GRS's success is strongly correlated to maximizing the total number of group members gratified with the recommended items, which is as significant as producing items tailored to each user (Seo et al., 2018). Therefore, the maximum number of users should agree with the recommended items, where the degree of agreement on an item can be defined by the distribution of the ratings that the item received from group members. Concretely, in the case of even distribution, the agreement level within the group is considered low. However, if the distribution is unimodal, this may indicate that a consensus among the vast majority of the group members is reached. It is safe to add items providing consensus to the recommendation list in the presence of agreement.

A few recent studies have utilized the standard deviation to analyze how users' ratings in a group spread out, which is currently the de facto standard to measure the dispersion of ratings in GRSs (Seo et al., 2018). In doing so, items with high deviations are associated with disagreements among group members and filtered out from the recommendations list. As long as the ratings have unimodal distribution, it is wise to employ the standard deviation for detecting such disagreements. However, when the ratings demonstrate a multi-modal distribution, the standard deviation may not be ideal for analyzing users' conflict.

A multi-modal distribution of ratings can be observed when groups in a GRS consist of subgroups of users with very similar preferences. A subgroup within a group may be highly pleased with an item in these cases, whereas another subgroup may dislike it. Thus, the users in the first subgroup provide high ratings, while those in the latter provide low ratings. Worse, if the rating scale is large enough, e.g., a ten-star rating scale, and/or if the groups are large, it is likely to observe more than two subgroups. In the case of MovieLens dataset,[1] for instance, there exist various items whose ratings show a multi-modal distribution. More specifically, the relative frequencies of the ratings in the [1-5] scale provided for the 853rd and 854th items in the dataset are {0.31, 0, 0, 0.38, 0.31} and {0.21, 0, 0, 0.29, 0.5}, respectively.

When the item's distribution is multi-modal, its standard deviation is high indeed, which may lead to the item being removed from the recommendations list. However, there can still be a consensus on the item with different levels. To deal with this problem, we suggest utilizing entropy in analyzing the distributions of ratings.

Entropy has its roots in thermodynamic, though, it was Shannon who adopted the concept of entropy to measure uncertainty for information theory (Shannon, 1948). Since then, Shannon's entropy (or simply entropy) has found a wealth of applications in information theory and computer science to quantify uncertainty/randomness in a system. In recommender systems, entropy has been successfully used and preferred over the standard deviation (Kaleli, 2014; Yargic & Bilge, 2019) since it is more robust than the standard deviation in analyzing distributions of ratings, especially when a multi-modal distribution has occurred. What is more, the standard deviation is for continuous variables, but usually, the ratings are discrete by its nature, which calls for employing entropy.

For the reasons mentioned above, we propose to incorporate the *hybridized* techniques (i.e., AU_{AV} and AV_{AU}), along with entropy. We shall name this final aggregation technique as *Agreement without Uncertainty*, AwU in short. Before proceeding with the details of the AwU, we give brief information on how entropy is considered within the scope of a rating system below.

Suppose that there is a set of possible ratings $R = r_1, r_2, \ldots, r_k$ with probabilities $P = p_1, p_2, \ldots, p_k$ for a rating vector v, which includes the preferences of group members for an item. Shannon entropy of such vector is denoted as $H(v)$ and calculated as in Eq. (3).

$$H(v) = -\sum_{i=1}^{k} p(r_i) log_2(p(r_i)) \tag{3}$$

Note that entropy as calculated in Eq. (3) indicates the degree of uncertainty of an item. Concretely, higher $H(v)$ means that users are uncertain with the item, which is to say that there is nearly no agreement among the users; lower $H(v)$, on the other hand, indicates the users have reached a certain level of consensus about the item.

Algorithm 1 summarizes the proposed AwU technique. Accordingly, given a rating-based user–item matrix of a group ($R_{U\times I}$) and threshold coefficient (τ), the AwU first calculates the entropy for all items using Eq. (3) according to the ratings given by members in the group for the corresponding item (*lines 2–4*). Once the algorithm calculates the entropy values, it then normalizes them so that into [0, 1] interval by min–max normalization (*line 5*). After that, the algorithm subtracts the normalized entropy values from 1 to compute information gain quantifying the amount of consensus among the group members (*lines 6–8*). Finally, the AwU estimate group ratings using the *hybridized* techniques for items only if the associated information gain is above the multiplication of the pre-defined threshold (τ) with the average gain of the group (m) (*lines 10–15*). In other words, this technique disregards items causing high entropy, which is an indicator of the lack of disagreement among users and does not count them as recommendable items.

4.4. Illustrative example

In this section, we provide a simple example to clarify how the proposed AwU technique estimates group ratings in the aggregation phase. To this end, we define a sample group that consists of seven members who provide preferences in the $[1-5]$ scale for six items, as presented in Table 11.

The AwU technique initially calculates an entropy value for each item, as in Eq. (3) using corresponding users' preferences and then

[1] http://www.grouplens.org/.

Algorithm 1 The AwU technique

Input: Rating matrix of the group g ($R_{U\times I}$), threshold value (τ)
Initialize:
$e_{(1\times I)} \leftarrow null$ ▷ *entropy vector of items*
Calculate entropy and information gain of each item:
for i in {1, 2, ⋯, I} **do**
  $e(i) \leftarrow$ H(R(:,i)) ▷ *using Eq. (3)*
end for
$\overline{e} \leftarrow$ NORMALIZE(e) ▷ *by min–max normalization*
for i in {1, 2, ⋯, I} **do**
  $\overline{e}(i) \leftarrow 1 - \overline{e}(i)$ ▷ *information gain*
end for
$m \leftarrow$ MEAN($\overline{e}$) ▷ *mean information gain*
Estimate group ratings for g:
for i in {1, 2, ⋯, I} **do**
  if $\overline{e}(i) > \tau \times m$ **then** ▷ *eliminate items with high entropy*
    $R^{AU}_{g,i} \leftarrow \mathrm{AU_{AV}(i)}$ ▷ *using Eq. (1)*
    $R^{AV}_{g,i} \leftarrow \mathrm{AV_{AU}(i)}$ ▷ *using Eq. (2)*
  end if
end for
return $R^{AU}_{g,i}$ or $R^{AV}_{g,i}$

Table 11
The example group with its members' preferences to items.

	i_1	i_2	i_3	i_4	i_5	i_6
u_1	1	5	⊥	1	⊥	3
u_2	2	⊥	⊥	2	5	3
u_3	4	⊥	5	4	4	⊥
u_4	3	5	5	5	⊥	3
u_5	4	5	2	4	⊥	⊥
u_6	⊥	4	2	4	1	⊥
u_7	2	5	5	⊥	3	⊥

Table 12
The entropy and information gain of the items.

	i_1	i_2	i_3	i_4	i_5	i_6
Entropy	1.92	0.72	0.97	1.79	2	0
$Entropy_{normalized}$	0.96	0.36	0.48	0.89	1	0
Information gain	0.04	0.64	0.52	0.10	0	1

normalizes these entropy values using the min–max normalization to force them to lie in the range $[0, 1]$. Having done that, it computes the information gain by subtracting the entropy from 1 for each item to measure the degree of the agreement provided by group members on the items, as presented in Table 12. In the example, based on the computed gain values, i_6 can be considered the item on which the group members reach the highest consensus, as it associates with the minimum entropy value (max. information gain). On the other hand, i_5 can be considered the item on which there is nearly no agreement among the group members, as it associates with the highest entropy score, thus the lowest information gain.

Having computed gain values for all items, the AwU calculates the average gain of the group (m), which is used in determining the threshold value. The average gain of the group can also indicate the degree of consensus reached by all group members. Concretely, a high value of m demonstrates that the group consists of users having very similar tastes, which is to say the group is well-constructed; a low value of m, on the other hand, means that the group contains individuals with different tastes. Going back to the example, the average gain m for the group is equal to 0.383, which indicates a moderate consensus among the users.

At the final step, the AwU calculates group ratings using one of the *hybridized* techniques presented in Eqs. (1) and (2) only for items whose gain is above a threshold value obtained by multiplying m with a predefined coefficient (τ). Here, τ regulates the effect of entropy in removing items from the recommendation list. Specifically, when τ is set to zero, then entropy has no impact on the *hybridized* techniques, as no item is eliminated in this case, whereas when it is set to one; then it has the maximum impact leading the *hybridized* techniques to deal with fewer items. Based on the gain values presented in Table 12, the final group ratings calculated with the AwU technique are presented in Table 13, where τ is set to 0.5. Clearly from Table 12, the gain scores of i_1, i_4, and i_5 are lower than the value of $m\times\tau$, which is equivalent to $0.383\times0.5 = 0.191$. Thus, the AwU technique regards these items as not recommendable for the group, eliminating them as can be followed in Table 13. On the other hand, for the remaining items, i.e., i_2, i_3, and i_6, it calculates a group rating using either AV_{AU} or AU_{AV} techniques formulated in Eqs. (1) and (2), respectively. Then the GRS considers them in constructing a list of recommended items to the group.

Table 13
Group ratings calculated by the AwU technique.

		i_1	i_2	i_3	i_4	i_5	i_6
Group ratings	AU_{AV}	–	1.67	1.11	–	–	0
	AV_{AU}	–	1.33	1.11	–	–	0

Table 14
Statistics of utilized datasets.

Dataset	#Users	#Items	#Ratings	Density (%)
MLP	943	1682	100,000	6.3
MLM	6040	3952	1,000,000	4.25
NF	10,000	17,700	2,337,295	1.32

5. Experimental studies

In this section, we scrutinize the performance of the proposed aggregation techniques in terms of their accuracy, fairness, and satisfaction on real-world datasets and then discuss the insights gained from the empirical outcomes.

5.1. Datasets and evaluation metrics

In the experiments, we make use of the famous MovieLens dataset that is publicly available, thanks to the GroupLens research team.[1] MovieLens comes with two variants based on the number of ratings included. These variants are named as MovieLens Public with 100K ratings, and MovieLens Million with 1M ratings. The former and the latter are abbreviated as MLP and MLM, respectively. Netflix[2] is another dataset of 100,480,507 ratings that 480,189 users gave to 17,770 movies. During the experiments, a subset of the Netflix prize dataset (NF) is also utilized, where 10,000 users are sampled, representing the density characteristics of both users and items in the original dataset. MLP, MLM, and NF datasets consist of user preferences on movies presented with discrete ratings on a five-star rating scale. Also, Table 14 provides detailed information about these datasets. Here, it is worth noting that they are large and extremely sparse datasets, where especially NF has only around one percent of all possible ratings.

To examine how accurate the proposed aggregation techniques, we employ the normalized Discounted Cumulative Gain (nDCG) metric, widely used in studies on group recommendation (Boratto et al., 2016; Masthoff, 2015). The nDCG measures the degree of quality of the recommended items by considering their actual ratings as well as their positions in the list of recommendations.

Assume that u is a user in a group g, and $r_{u,i}$ denotes the actual rating of u for item i. Also assume that the employed GRS recommends the

[1] http://www.grouplens.org/.
[2] http://www.netflixprize.com/.

following N items $\{i_1, i_2, \ldots, i_N\}$ to group g. In this case, the Discounted Cumulative Gain (DCG) and the normalized Discounted Cumulative Gain (nDCG) for user u are calculated using the formulas given in Eqs. (4) and (5), respectively.

$$\mathrm{DCG}_N^u = r_{u,i_1} + \sum_{n=2}^{N} \frac{r_{u,i_n}}{log_2(n)} \tag{4}$$

$$n\mathrm{DCG}_N^u = \frac{\mathrm{DCG}_N^u}{\mathrm{IDCG}_N^u} \tag{5}$$

where IDCG_N^u denotes the maximum amount of possible gain for u, which can be calculated by re-ordering of N items so that it will be the ideal order for user u.

GRSs usually give rise to fairness issues, as such systems contain multiple stakeholders (Burke, 2017). Considering that the proposed aggregation techniques aim at satisfying group members evenly, it is necessary to examine their fairness performance. To this end, we utilize *m-proportionality* that interprets fairness as the share of group members u_i with at least m items in the recommended N items for which u_i has a high preference (Felfernig et al., 2018; Serbos et al., 2017). In this context, *m-proportionality*-based fairness metric measures how fair a list of items N is for the entire group g using the formula given in Eq. (6).

$$fairness_{m-prop}(g) = \frac{|g_N|}{|g|} \tag{6}$$

where g_N indicates the set of group members for whom the *m-proportionality* condition holds.

To comprehensively evaluate the quality of a recommendation list produced to a group g, we also employ a Group Satisfaction Metric (GSM) to measure the degree of satisfaction based on each member (Barzegar Nozari & Koohi, 2020). The GSM metric can be calculated using the formula given in Eq. (7). Note that we assign 3.5 as the threshold value for determining if a group member is satisfied in estimating both fairness and GSM metrics since positive votes intuitively correspond to 4 and 5 for a five-star rating scale (Bobadilla et al., 2010).

$$\mathrm{GSM}_g = \frac{\sum_{u \in g} |I_u \cap N|}{|g| \times |N|} \tag{7}$$

where I_u demonstrates the set of items that gratify user u.

All utilized metrics, i.e., nDCG, fairness, and GSM, require to know in advance the actual votes of group members for the recommended list of items. However, it is a fact that there is no such dataset containing ground truth values for group recommendations, as groups of users are not predefined in general (Boratto et al., 2016). Moreover, the datasets used in recommender systems are usually sparse; take MovieLens and Netflix as examples, thus having actual ratings of individuals is impossible most of the time. To handle this problem, we estimate members' actual ratings by a user-based collaborative filtering algorithm (Herlocker et al., 1999), as in Sacharidis (2019). This way, we obtain ground truth values for each group member, which is later used to determine non-real group ratings and compare them with a list of recommended items.

5.2. Benchmark methods

In the experimental studies, we consider AU, Mul, AV, Avg, AwM, BC, CR, LM, SC, and MP methods as the baseline aggregation techniques. Additionally, we select two improved aggregation approaches as the benchmark methods, UL (Seo et al., 2018) and IBGR (Barzegar Nozari & Koohi, 2020), to evaluate the performance of our proposed ultimate AwU technique more comprehensively.

Specifically, UL is an enhanced aggregation method that considers the distribution of the preferences given by group members during the aggregation process. To this end, it calculates deviations of the preferences provided for an item and then combines it with group scores computed with Avg and AV techniques to estimate ultimate group ratings for the corresponding item.

IBGR, on the other hand, has been introduced as a group recommendation method where social relationships and the influence of group members are considered significant elements in the aggregation process. More specifically, this method initially computes the group members' influence on each other based on similarity and trust, then utilizes them to determine the leaders trusted more than other members. It finally weights individual preferences with the Leaders' impact on other members to achieve group scores.

5.3. Experimentation methodology

To evaluate the proposed group recommendation scheme's performance, we perform a five-fold cross-validation procedure in this study. To this end, we divide the set of items into five subsets randomly, such that each subset consists of around 20% of the items. At each iteration, we employ one of the subsets as the test set, which serves as a set of test items and utilizes the remaining subsets as the training set. The training items are used to identify groups, while the test items are employed to test the performance of the proposed aggregation techniques and the baseline techniques explained below.

We utilize the k-means clustering algorithm to construct groups of different numbers. Concretely, we allow the number of groups, denoted as k, to range from 4 to 128, aiming to examine the effect of group size on the proposed aggregation techniques. For each number of groups, we compare the proposed techniques with the aggregation methods: AU, Mul, AV, Avg, AwM, BC, CR, LM, SC, MP, UL, and IBGR, ensuring that each category of the aggregation techniques is represented in the experiments.

When it comes to parameter settings of the aggregation techniques used in the experiments, the threshold value is selected as 3 for the techniques that employ AV, (i.e., AV itself, UL, our *hybridized* techniques AU_{AV} and AV_{AU}). It is because the positive ratings correspond to 4 and 5 for MovieLens dataset (Bobadilla et al., 2010). The threshold value for AwM technique is selected as 3, as well. Also, UL requires to tune weights of its elements Avg, AV, and SD carefully. In our experiments, we set the corresponding weights as $0.4, 0.2$, and 0.4, respectively, since the creators of UL experimentally show that it is one of the optimal settings for providing group recommendations of high-quality (Seo et al., 2018). Finally, we consider τ values ranging from 0 to 1 while performing the proposed *AwU* technique to see the effects of different threshold values.

We predict a group rating for each item from the test set using an aggregation technique in recommending items to a group. We sort these ratings in decreasing order; and then select top-N items based on their group ratings as a list of recommendations, where N is set to $1, 3, 5$, and 10. Finally, to evaluate how qualified the recommended top-N items concerning the group's actual preferences, we calculate the nDCG, fairness, and GSM values of the group members, and then average them.

5.4. Experimental results

5.4.1. Evaluation of the hybridized technique

To investigate the performance of the proposed *hybridized* aggregation techniques (i.e., AV_{AU} and AU_{AV}) on predicting group ratings, we conducted a broad set of experiments with different parameters, including the number of groups (k) and the size of recommendation list (N). Furthermore, we compared the outcomes of these experiments against 10 baseline aggregation techniques for MLP, MLM and NF datasets, as presented in Tables 15–17, respectively.

Tables 15, 16, and 17 show that BC, AU, and AV techniques exhibit relatively better performance in comparison with the other baseline techniques based on the nDCG results obtained from both datasets. This observation may be used to support our initial claim that AU and

Table 15
*n*DCG results for MLP dataset.

Top-N	Aggregation technique	Number of groups (k)					
		4	8	16	32	64	128
1	AU	0.823	0.816	0.817	0.805	0.813	0.808
	Mul	0.766	0.765	0.773	0.764	0.781	0.779
	AV	0.822	0.814	0.811	0.813	0.814	0.803
	Avg	0.766	0.768	0.773	0.766	0.782	0.780
	AwM	0.766	0.768	0.773	0.766	0.782	0.780
	BC	0.826	0.811	0.808	0.809	0.816	0.814
	CR	0.791	0.792	0.754	0.758	0.745	0.723
	LM	0.766	0.765	0.773	0.765	0.782	0.779
	SC	0.820	0.804	0.799	0.781	0.775	0.763
	MP	0.753	0.757	0.768	0.762	0.780	0.779
	$\boldsymbol{AV_{AU}}$	0.838*	0.818	0.816	0.821*	0.820*	0.817
	$\boldsymbol{AU_{AV}}$	0.844*	0.833*	0.828*	0.823*	0.820*	0.818*
3	AU	0.812	0.800	0.799	0.803	0.797	0.798
	Mul	0.761	0.768	0.767	0.776	0.783	0.784
	AV	0.819	0.811	0.806	0.810	0.797	0.795
	Avg	0.763	0.773	0.768	0.778	0.787	0.786
	AwM	0.761	0.771	0.768	0.778	0.786	0.785
	BC	0.818	0.805	0.803	0.805	0.803	0.804
	CR	0.784	0.778	0.742	0.747	0.731	0.709
	LM	0.761	0.769	0.767	0.776	0.784	0.784
	SC	0.805	0.789	0.785	0.782	0.773	0.766
	MP	0.754	0.762	0.762	0.770	0.781	0.784
	$\boldsymbol{AV_{AU}}$	0.821	0.815	0.811	0.813	0.806	0.803
	$\boldsymbol{AU_{AV}}$	0.833*	0.828*	0.821*	0.818*	0.816*	0.812*
5	AU	0.803	0.792	0.795	0.792	0.787	0.787
	Mul	0.760	0.768	0.764	0.773	0.777	0.776
	AV	0.809	0.805	0.797	0.798	0.785	0.782
	Avg	0.765	0.776	0.767	0.777	0.780	0.781
	AwM	0.762	0.773	0.766	0.776	0.778	0.778
	BC	0.808	0.800	0.798	0.796	0.792	0.792
	CR	0.773	0.770	0.742	0.735	0.718	0.691
	LM	0.760	0.769	0.765	0.774	0.778	0.777
	SC	0.795	0.782	0.779	0.774	0.764	0.757
	MP	0.752	0.761	0.761	0.768	0.773	0.777
	$\boldsymbol{AV_{AU}}$	0.814	0.808	0.801	0.802	0.792	0.789
	$\boldsymbol{AU_{AV}}$	0.828*	0.824*	0.814*	0.810*	0.805*	0.801*
10	AU	0.792	0.787	0.788	0.785	0.782	0.774
	Mul	0.757	0.767	0.768	0.772	0.775	0.767
	AV	0.801	0.797	0.792	0.783	0.776	0.757
	Avg	0.775	0.780	0.777	0.781	0.784	0.774
	AwM	0.768	0.775	0.775	0.775	0.778	0.766
	BC	0.798	0.792	0.792	0.787	0.785	0.778
	CR	0.762	0.764	0.737	0.720	0.704	0.661
	LM	0.759	0.768	0.769	0.773	0.777	0.768
	SC	0.785	0.777	0.775	0.769	0.763	0.751
	MP	0.748	0.760	0.764	0.769	0.775	0.769
	$\boldsymbol{AV_{AU}}$	0.804	0.799	0.795	0.787	0.782	0.763
	$\boldsymbol{AU_{AV}}$	0.819*	0.815*	0.811*	0.803*	0.800*	0.787*

*For significance at 95%; w.r.t. the best-performing baseline technique.

Table 16
*n*DCG results for MLM dataset.

Top-N	Aggregation technique	Number of Groups (k)					
		4	8	16	32	64	128
1	AU	0.836	0.837	0.824	0.825	0.824	0.821
	Mul	0.738	0.746	0.754	0.759	0.764	0.768
	AV	0.839	0.838	0.828	0.831	0.826	0.824
	Avg	0.745	0.746	0.754	0.759	0.764	0.768
	AwM	0.738	0.746	0.754	0.759	0.764	0.768
	BC	0.831	0.832	0.835	0.828	0.829	0.827
	CR	0.819	0.821	0.814	0.793	0.789	0.793
	LM	0.738	0.746	0.754	0.759	0.764	0.768
	SC	0.834	0.828	0.815	0.813	0.810	0.803
	MP	0.755	0.761	0.761	0.762	0.764	0.768
	$\boldsymbol{AV_{AU}}$	0.839	0.839	0.831	0.832	0.830	0.828
	$\boldsymbol{AU_{AV}}$	0.849*	0.844*	0.839*	0.839*	0.836*	0.834*
3	AU	0.835	0.817	0.822	0.814	0.810	0.811
	Mul	0.749	0.753	0.756	0.758	0.764	0.773
	AV	0.839	0.823	0.827	0.819	0.816	0.812
	Avg	0.750	0.757	0.757	0.759	0.764	0.773
	AwM	0.750	0.753	0.756	0.758	0.764	0.773
	BC	0.840	0.821	0.826	0.819	0.814	0.814
	CR	0.768	0.783	0.770	0.776	0.772	0.772
	LM	0.750	0.753	0.756	0.758	0.764	0.773
	SC	0.828	0.808	0.811	0.803	0.798	0.798
	MP	0.761	0.762	0.765	0.767	0.769	0.774
	$\boldsymbol{AV_{AU}}$	0.840	0.826	0.828	0.822	0.818	0.817
	$\boldsymbol{AU_{AV}}$	0.850*	0.833*	0.836*	0.830*	0.826*	0.825*
5	AU	0.830	0.813	0.816	0.810	0.805	0.806
	Mul	0.750	0.755	0.757	0.760	0.765	0.775
	AV	0.833	0.819	0.822	0.816	0.810	0.808
	Avg	0.756	0.760	0.759	0.761	0.766	0.775
	AwM	0.751	0.755	0.757	0.760	0.765	0.775
	BC	0.833	0.818	0.820	0.814	0.809	0.810
	CR	0.769	0.775	0.771	0.774	0.767	0.769
	LM	0.751	0.755	0.757	0.760	0.765	0.775
	SC	0.822	0.803	0.806	0.798	0.794	0.794
	MP	0.762	0.760	0.763	0.766	0.770	0.776
	$\boldsymbol{AV_{AU}}$	0.836	0.822	0.824	0.818	0.813	0.813
	$\boldsymbol{AU_{AV}}$	0.847*	0.830*	0.832*	0.827*	0.823*	0.821*
10	AU	0.821	0.819	0.812	0.806	0.805	0.803
	Mul	0.756	0.758	0.762	0.768	0.773	0.776
	AV	0.825	0.826	0.819	0.810	0.809	0.806
	Avg	0.770	0.766	0.765	0.770	0.774	0.777
	AwM	0.757	0.759	0.763	0.767	0.773	0.776
	BC	0.823	0.823	0.817	0.809	0.808	0.806
	CR	0.789	0.789	0.783	0.774	0.769	0.767
	LM	0.756	0.759	0.762	0.768	0.773	0.776
	SC	0.813	0.809	0.801	0.795	0.796	0.791
	MP	0.756	0.765	0.767	0.769	0.776	0.779
	$\boldsymbol{AV_{AU}}$	0.828	0.828	0.821	0.813	0.812	0.810
	$\boldsymbol{AU_{AV}}$	0.836*	0.839*	0.830*	0.824*	0.823*	0.819*

*For significance at 95%; w.r.t. the best-performing baseline technique.

AV are ideal choices for base aggregation techniques in constructing a hybridized technique. Also, one may wish to employ BC as well aggregation technique relying on its high performance shown in Tables 15–17, but the fact that it is a computationally intensive technique due the sorting process it involves can make it undesirable. Moreover, it calculates group scores that rely on the rankings of items, making it challenging to incorporate the BC into a hybridized technique.

The outcomes of the experiments conducted on both datasets also indicate that AU, AV, BC, CR, and SC techniques perform better as group size increases. On the other hand, the other baseline techniques achieve their best in the presence of relatively small groups. Also, almost all aggregation techniques seem to be less significant considering the decreasing *n*DCG scores, as the recommendation list (i.e., N) grows. This finding can be attributed to the following reason. We estimate the ground truths to calculate *n*DCG scores with a collaborative filtering algorithm. When we recommend a large number of items using an aggregation technique, we rely on the *estimated* ratings to a more considerable extent to calculate the corresponding *n*DCG score. In this case, *n*DCG scores may be misleading.

When it comes to comparing the MLP, MLM, and the NF dataset results, the aggregation techniques are more successful on the MLM dataset. Here, the main reason could be that in MLM, the total number of ratings that an item has received outnumbers the one in both MLP and NF, which makes the aggregation technique used is more robust, as more ratings are involved in the aggregation. This holds true for both AV_{AU} and AU_{AV} techniques as well as for the other baseline aggregation techniques.

The results presented in Tables 15, 16, and 17 show that the *hybridized* aggregation techniques usually outperform all of the baseline techniques concerning all datasets. We also performed one-tailed t-tests to ensure whether the improvements are statistically significant at a 95% confidence level. The proposed hybridized technique AU_{AV} was found to be significantly better than both AV_{AU}. Therefore, it is concluded that utilizing AU as the decisive factor in an aggregation

Table 17
*n*DCG results for NF dataset.

Top-N	Aggregation technique	Number of groups (k)					
		4	8	16	32	64	128
1	AU	0.770	0.776	0.772	0.773	0.770	0.774
	Mul	0.741	0.734	0.740	0.744	0.753	0.755
	AV	0.784	0.785	0.782	0.782	0.779	0.781
	Avg	0.741	0.734	0.740	0.744	0.753	0.755
	AwM	0.741	0.734	0.740	0.744	0.753	0.755
	BC	0.782	0.787	0.780	0.785	0.779	0.780
	CR	0.764	0.764	0.772	0.767	0.767	0.760
	LM	0.741	0.734	0.740	0.744	0.753	0.755
	SC	0.736	0.737	0.745	0.749	0.751	0.750
	MP	0.736	0.730	0.726	0.736	0.742	0.746
	$\boldsymbol{AV_{AU}}$	0.784	0.786	0.786	0.786	0.787	0.786
	$\boldsymbol{AU_{AV}}$	0.796*	0.799*	0.795*	0.795*	0.794*	0.791*
3	AU	0.762	0.761	0.765	0.764	0.766	0.769
	Mul	0.740	0.734	0.744	0.745	0.751	0.753
	AV	0.772	0.771	0.777	0.775	0.775	0.776
	Avg	0.740	0.734	0.744	0.745	0.751	0.753
	AwM	0.740	0.734	0.744	0.745	0.751	0.753
	BC	0.772	0.777	0.771	0.772	0.773	0.775
	CR	0.755	0.756	0.758	0.752	0.749	0.747
	LM	0.740	0.734	0.744	0.745	0.751	0.753
	SC	0.741	0.745	0.747	0.750	0.751	0.752
	MP	0.739	0.730	0.737	0.742	0.747	0.752
	$\boldsymbol{AV_{AU}}$	0.777	0.777	0.778	0.778	0.779	0.779
	$\boldsymbol{AU_{AV}}$	0.793*	0.793*	0.785*	0.786*	0.788*	0.786*
5	AU	0.762	0.758	0.763	0.763	0.765	0.766
	Mul	0.741	0.737	0.745	0.746	0.752	0.754
	AV	0.773	0.771	0.773	0.773	0.774	0.774
	Avg	0.741	0.737	0.745	0.746	0.752	0.754
	AwM	0.741	0.737	0.745	0.746	0.752	0.754
	BC	0.772	0.774	0.769	0.771	0.772	0.772
	CR	0.759	0.755	0.755	0.748	0.748	0.745
	LM	0.741	0.737	0.745	0.746	0.752	0.754
	SC	0.745	0.744	0.750	0.749	0.751	0.752
	MP	0.739	0.731	0.741	0.743	0.750	0.754
	$\boldsymbol{AV_{AU}}$	0.777	0.775	0.776	0.777	0.777	0.777
	$\boldsymbol{AU_{AV}}$	0.789*	0.791*	0.785*	0.785*	0.786*	0.783
10	AU	0.761	0.760	0.761	0.762	0.765	0.765
	Mul	0.744	0.740	0.747	0.748	0.754	0.756
	AV	0.770	0.771	0.771	0.772	0.772	0.771
	Avg	0.744	0.740	0.747	0.748	0.754	0.756
	AwM	0.744	0.740	0.747	0.748	0.754	0.756
	BC	0.771	0.773	0.767	0.769	0.770	0.770
	CR	0.757	0.752	0.750	0.746	0.746	0.740
	LM	0.744	0.740	0.747	0.748	0.754	0.756
	SC	0.747	0.746	0.750	0.750	0.752	0.752
	MP	0.742	0.736	0.745	0.747	0.754	0.756
	$\boldsymbol{AV_{AU}}$	0.772	0.775	0.774	0.775	0.775	0.774
	$\boldsymbol{AU_{AV}}$	0.788*	0.789*	0.785*	0.784*	0.785*	0.782*

*For significance at 95%; w.r.t. the best-performing baseline technique.

technique of hybridized type is a better choice than employing AV in terms of providing group recommendations of higher quality.

5.4.2. Evaluation of the AwU technique

Based on the experiments presented in the previous subsection, we concluded that between the two *hybridized* aggregation techniques proposed, AU_{AV} is superior to AV_{AU}. We also witnessed that AU_{AV} outperforms the other baseline techniques considered. We investigate if we can further improve group recommendations in case we eliminate items through the entropy measure before employing AU_{AV}. Recall that we call the approach of decorating AU_{AV} with the entropy as AwU.

To examine the effectiveness of the AwU technique, we conducted several experiments varying the parameters such as the number of groups, the number of recommended items, and τ determining entropy's influence level.

Figs. 2, 3, and 4 present the *n*DCG scores of the AwU on the MLP, MLM, and NF datasets, respectively. Based on the positive trend in the *n*DCG scores in almost all settings, we claim with confidence that considering the entropy of rating distributions enhances the performance of AU_{AV} in nearly all schemes, which means the AwU outperforms AU_{AV} in general. Such observation becomes more apparent when the number of groups is relatively small, meaning that the groups are large.

Ideally, τ should be set to a value between 0.8 and 1, relying on the experiments we carried on the MLP, MLM, and NF datasets, which can be evidence that there is a need for employing the entropy. In particular, we found that when τ is set to 0.8 for both MLP and NF, AwU is significantly better than AU_{AV} at 95% confidence level when the number of groups is 4, 8, and 16. This finding also holds for the different numbers of recommended items. Similarly, for the MLM dataset, AwU performs better than AU_{AV}, and the enhancements are statistically significant at a 95% confidence level for all numbers of groups but 128.

We also perform various additional experiments to compare the *n*DCG performance of the AwU technique against both benchmark approaches (i.e., UL and IBGR) for MLP, MLM, and NF datasets, as presented in Tables 18, 19, and 20, respectively. According to the obtained results, it can be concluded that our ultimate AwU technique significantly outperforms both benchmarks, as well. This finding is more apparent in the presence of the medium and large groups, which is parallel with the outcomes of the experiments performed in the previous section.

We conclude this part by highlighting the following remark. When the number of groups is small for large datasets, this means that the groups are crowded, which makes it possible that the groups consist of members with diverse opinions on some items. These items can be associated with high entropy values, so it is wise to disregard items with high entropy before the aggregation step. We observed that even the performance of AU_{AV}, which is the most successful among other baseline aggregation techniques, is improved through the utilization of entropy.

Table 18
*n*DCG comparison of the AwU technique against benchmarks for MLP.

Top-N	Aggregation method	Number of groups (k)					
		4	8	16	32	64	128
1	UL	0.830	0.819	0.812	0.814	0.814	0.805
	IBGR	0.821	0.817	0.810	0.810	0.803	0.801
	AwU ($\tau = 0.8$)	0.858*	0.843*	0.834*	0.824*	0.820*	0.816*
3	UL	0.822	0.814	0.808	0.811	0.803	0.799
	IBGR	0.818	0.810	0.808	0.807	0.802	0.795
	AwU ($\tau = 0.8$)	0.851*	0.843*	0.827*	0.819	0.815	0.810*
5	UL	0.811	0.807	0.801	0.802	0.793	0.790
	IBGR	0.808	0.806	0.796	0.795	0.788	0.781
	AwU ($\tau = 0.8$)	0.840*	0.834*	0.821*	0.810*	0.805*	0.799
10	UL	0.802	0.799	0.797	0.794	0.790	0.778
	IBGR	0.799	0.789	0.786	0.788	0.779	0.771
	AwU ($\tau = 0.8$)	0.831*	0.822*	0.814*	0.803*	0.799	0.783

*For significance at 95%; w.r.t. the best-performing benchmark method.

5.4.3. Fairness and satisfaction analysis

In this section, we perform various additional experiments to provide a comprehensive analysis of the performance of our ultimate AwU technique in terms of fairness and satisfaction. In these experiments, we consider three baseline aggregation techniques, AU, AV, and BC, as they are the best-performing ones according to the outcomes presented in Section 5.4.1. Also, we compare our ultimate AwU technique against two state-of-the-art approaches, which are UL and IBGR. In the experiments, we consider all three group formations, i.e., small, medium, and large, by selecting k as 4, 16, and 64, respectively. Finally, we vary m values from 1 to 5 to investigate the effect of the m value on the fairness metric.

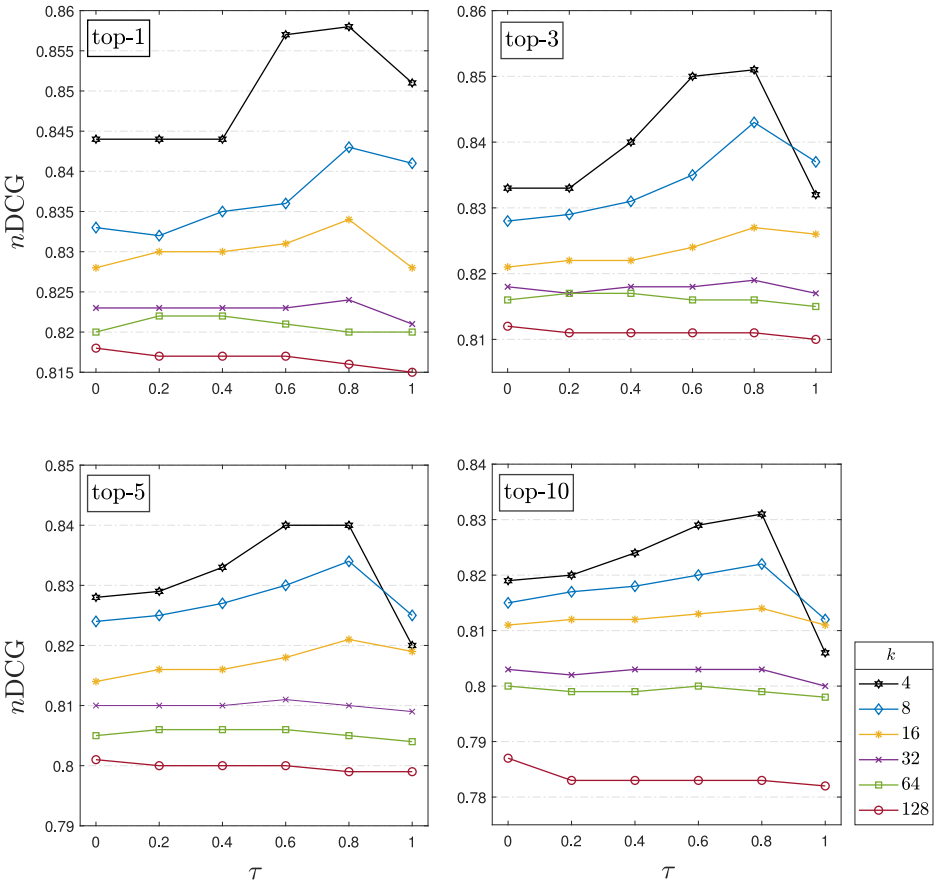

Fig. 2. *n*DCG results of the AwU technique for MLP.

Table 19
*n*DCG comparison of the AwU technique against benchmarks for MLM.

Top-N	Aggregation method	Number of groups (k)					
		4	8	16	32	64	128
1	UL	0.841	0.839	0.827	0.830	0.827	0.827
	IBGR	0.827	0.819	0.811	0.810	0.805	0.802
	AwU ($\tau = 0.8$)	0.861*	0.854*	0.847*	0.845*	0.839*	0.836
3	UL	0.840	0.824	0.827	0.820	0.816	0.814
	IBGR	0.829	0.826	0.816	0.816	0.805	0.805
	AwU ($\tau = 0.8$)	0.855*	0.844*	0.840*	0.833	0.828	0.825*
5	UL	0.835	0.821	0.823	0.817	0.812	0.810
	IBGR	0.821	0.819	0.813	0.808	0.799	0.793
	AwU ($\tau = 0.8$)	0.851*	0.840*	0.838*	0.834*	0.827*	0.822
10	UL	0.828	0.829	0.821	0.814	0.812	0.809
	IBGR	0.821	0.819	0.809	0.807	0.796	0.792
	AwU ($\tau = 0.8$)	0.844*	0.843*	0.835*	0.829*	0.826	0.820

*For significance at 95%; w.r.t. the best-performing benchmark method.

Table 20
*n*DCG comparison of the AwU technique against benchmarks for NF.

Top-N	Aggregation method	Number of Groups (k)					
		4	8	16	32	64	128
1	UL	0.784	0.785	0.785	0.782	0.782	0.782
	IBGR	0.781	0.781	0.779	0.775	0.776	0.774
	AwU ($\tau = 0.8$)	0.815*	0.798*	0.794*	0.794*	0.793*	0.792
3	UL	0.778	0.778	0.776	0.776	0.773	0.772
	IBGR	0.773	0.772	0.770	0.768	0.764	0.769
	AwU ($\tau = 0.8$)	0.811*	0.801*	0.790*	0.789	0.788	0.787
5	UL	0.776	0.775	0.775	0.774	0.774	0.774
	IBGR	0.770	0.771	0.769	0.770	0.768	0.767
	AwU ($\tau = 0.8$)	0.809*	0.798*	0.790*	0.788*	0.787	0.784
10	UL	0.775	0.774	0.774	0.773	0.773	0.771
	IBGR	0.773	0.773	0.771	0.770	0.770	0.768
	AwU ($\tau = 0.8$)	0.804*	0.795*	0.791*	0.787*	0.786*	0.782

*For significance at 95%; w.r.t. the best-performing benchmark method.

Tables 21, 22, and 23 present fairness results of the aggregation methods for top-5 group recommendations on MLP, MLM, and NF datasets, respectively. Based on the obtained results, it can be concluded that our ultimate AwU technique provides group recommendations with relatively more fairness in comparison with both baseline and benchmark methods. Such finding becomes more apparent when the groups are medium or large, which is similar to the trend in the *n*DCG results obtained from experiments in Section 5.4.1. We again conduct one-tailed t-tests to ensure whether the enhancements are statistically significant at 95% confidence level. In particular, we found that the AwU technique is significantly better than other methods when the groups are large or medium. Empirical outcomes also demonstrate that all techniques' fairness scores unsurprisingly decrease as the value of m increases since the number of group members satisfied with at least m items usually diminishes for larger m values.

Also, we compare the GSM results of the AwU technique against both baselines and benchmarks for MLP, MLM, and NF datasets, as presented in Table 24. According to the obtained results, it can be concluded that the AwU technique significantly improves the overall satisfaction of group members, especially when the groups are large or medium, as well.

5.5. Insights and discussions

In the present study, we aim at identifying common tastes of the users in a group to produce a list of recommended items that can

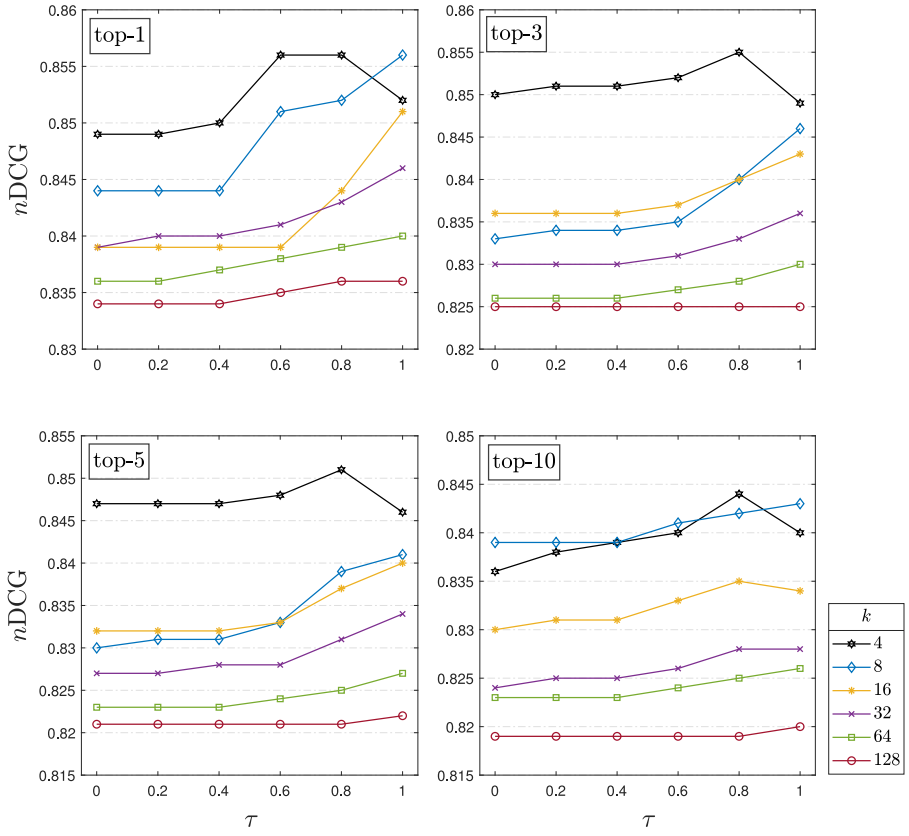

Fig. 3. *n*DCG results of the AwU technique for MLM.

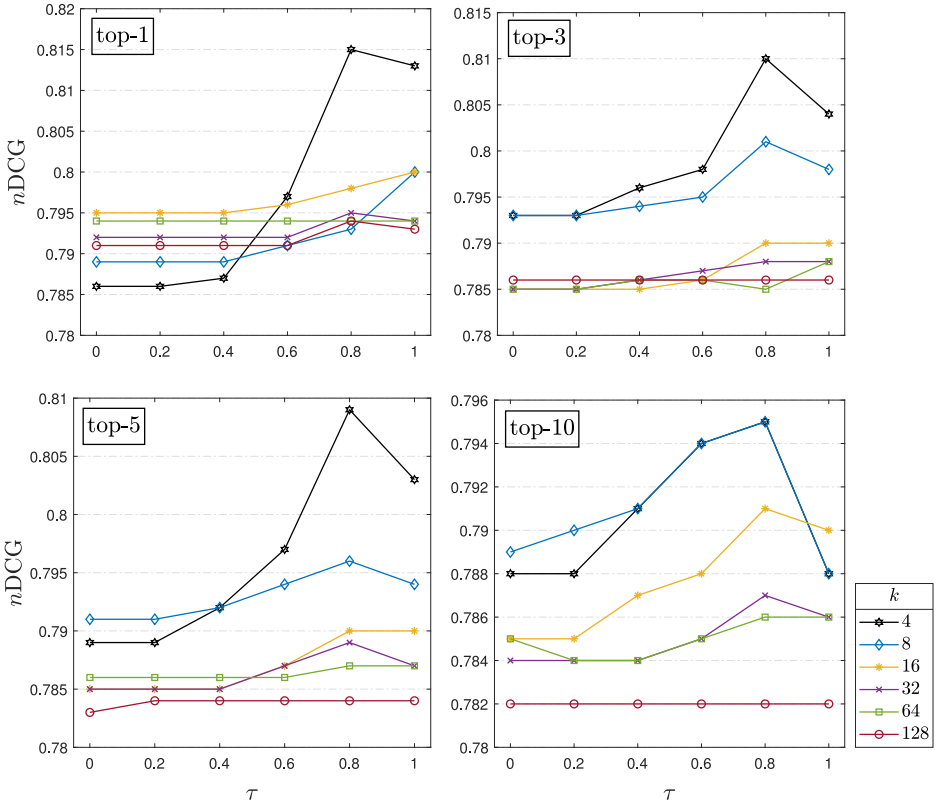

Fig. 4. *n*DCG results of the AwU technique for NF.

Table 21
Fairness results of top-5 group recommendations for MLP dataset.

Group	Aggregation method	m				
		1	2	3	4	5
Large (k = 4)	AU	0.793	0.625	0.448	0.278	0.111
	AV	0.830	0.645	0.473	0.303	0.144
	BC	0.817	0.643	0.462	0.309	0.142
	UL	0.828	0.646	0.473	0.302	0.149
	IBGR	0.821	0.651	0.468	0.312	0.152
	AwU (τ = **0.8**)	0.849*	0.706*	0.525*	0.352*	0.176*
Medium (k = 16)	AU	0.798	0.604	0.410	0.248	0.104
	AV	0.823	0.616	0.432	0.277	0.126
	BC	0.810	0.615	0.432	0.262	0.122
	UL	0.822	0.622	0.432	0.278	0.126
	IBGR	0.819	0.619	0.439	0.272	0.130
	AwU (τ = **0.8**)	0.840*	0.633*	0.459*	0.296*	0.142*
Small (k = 64)	AU	0.789	0.600	0.398	0.242	0.108
	AV	0.802	0.614	0.411	0.255	0.118
	BC	0.795	0.610	0.407	0.249	0.123
	UL	0.804	0.615	0.414	0.254	0.116
	IBGR	0.810	0.613	0.408	0.248	0.113
	AwU (τ = **0.8**)	0.802	0.611	0.420	0.268*	0.114

*For significance at 95%; w.r.t. the best-performing benchmark method.

Table 22
Fairness results of top-5 group recommendations for MLM dataset.

Group	Aggregation method	m				
		1	2	3	4	5
Large (k = 4)	AU	0.876	0.744	0.603	0.430	0.200
	AV	0.880	0.760	0.617	0.439	0.221
	BC	0.880	0.755	0.611	0.450	0.224
	UL	0.881	0.762	0.619	0.445	0.225
	IBGR	0.883	0.764	0.610	0.451	0.230
	AwU (τ = **0.8**)	0.908*	0.782*	0.637*	0.465*	0.244*
Medium (k = 16)	AU	0.863	0.730	0.561	0.389	0.195
	AV	0.869	0.737	0.582	0.398	0.203
	BC	0.868	0.730	0.580	0.391	0.203
	UL	0.869	0.738	0.584	0.403	0.206
	IBGR	0.865	0.743	0.581	0.413	0.210
	AwU (τ = **0.8**)	0.883*	0.762*	0.600*	0.430*	0.226*
Small (k = 64)	AU	0.855	0.703	0.541	0.364	0.179
	AV	0.865	0.715	0.547	0.377	0.190
	BC	0.859	0.711	0.544	0.376	0.191
	UL	0.866	0.722	0.548	0.378	0.191
	IBGR	0.871	0.730	0.550	0.371	0.188
	AwU (τ = **0.8**)	0.874	0.734	0.572*	0.390*	0.204*

*For significance at 95%; w.r.t. the best-performing benchmark method.

Table 23
Fairness results of top-5 group recommendations for NF dataset.

Group	Aggregation method	m				
		1	2	3	4	5
Large (k = 4)	AU	0.753	0.488	0.312	0.190	0.083
	AV	0.756	0.498	0.325	0.204	0.092
	BC	0.744	0.477	0.299	0.187	0.092
	UL	0.755	0.499	0.325	0.204	0.094
	IBGR	0.758	0.501	0.331	0.214	0.091
	AwU (τ = **0.8**)	0.774*	0.524*	0.344*	0.234*	0.117*
Medium (k = 16)	AU	0.745	0.468	0.302	0.192	0.089
	AV	0.751	0.484	0.315	0.196	0.094
	BC	0.744	0.467	0.300	0.196	0.094
	UL	0.750	0.486	0.318	0.201	0.095
	IBGR	0.754	0.484	0.322	0.204	0.094
	AwU (τ = **0.8**)	0.779*	0.507*	0.341*	0.226*	0.096
Small (k = 64)	AU	0.736	0.470	0.297	0.195	0.090
	AV	0.742	0.477	0.306	0.200	0.097
	BC	0.732	0.466	0.298	0.197	0.094
	UL	0.743	0.479	0.308	0.204	0.098
	IBGR	0.746	0.473	0.312	0.215	0.097
	AwU (τ = **0.8**)	0.743	0.475	0.303	0.199	0.096

*For significance at 95%; w.r.t. the best-performing benchmark method.

Table 24
GSM results of top-5 group recommendations.

Dataset	Aggregation technique	Group		
		Large (k = 4)	Medium (k = 16)	Small (k = 64)
MLP	AU	0.692	0.689	0.674
	AV	0.720	0.714	0.688
	BC	0.717	0.702	0.684
	UL	0.725	0.717	0.695
	IBGR	0.723	0.716	0.690
	AwU (τ = **0.8**)	0.767*	0.723*	0.698
MLM	AU	0.773	0.788	0.766
	AV	0.790	0.806	0.778
	BC	0.785	0.798	0.773
	UL	0.793	0.810	0.780
	IBGR	0.792	0.804	0.778
	AwU (τ = **0.8**)	0.826*	0.823*	0.785
NF	AU	0.680	0.677	0.668
	AV	0.708	0.701	0.669
	BC	0.712	0.709	0.674
	UL	0.715	0.708	0.684
	IBGR	0.718	0.711	0.683
	AwU (τ = **0.8**)	0.754*	0.718*	0.674

*For significance at 95%; w.r.t. the best-performing benchmark method.

satisfy the entire group as much as possible. With this in mind, we first propose two novel *hybridized* techniques AV_{AU}, and AU_{AV} that aggregate preferences of individuals in a group. We then propose using the entropy to detect items on which there is a disagreement among group members. As a product, we offer the AwU technique by leveraging the entropy calculation for AU_{AV}.

In the experiments, we observed that the aggregation techniques of hybridized type, AV_{AU}, AU_{AV} and UL, achieved the highest nDCG scores in comparison to the other baseline techniques, as we had anticipated at the outset. This is because employing multiple aggregation techniques aids in overcoming the shortcomings of the base techniques and handles the problem of combining individuals' preferences from multiple points of view. More specifically, AU_{AV} is superior to the other two *hybridized* technique AV_{AU} and UL. The point is that AU_{AV} is mainly dominated by AU, which assumes the items on which group members provided a consensus are more precious than the highly-rated items for having overall group satisfaction.

According to the experiments conducted on three datasets, AwU was found to be the most successful aggregation technique, especially when the groups are large. Considering the fact that AwU is the only technique that analyzes the distribution of the ratings through entropy, this result verifies analyzing the rating distribution becomes a critical element of the aggregation phase in GRS as the groups get crowded. Moreover, in AwU, the entropy effect is positively correlated with a parameter τ ranging from 0 to 1. The empirical outcomes also showed that AwU performs best when τ is set to a value between 0.8 and 1 again, which supports our claim that the entropy calculation plays a crucial role in having accurate group preferences.

To sum up, the experiments allow us to draw the following conclusion: the proposed aggregation techniques, AV_{AU}, and AU_{AV}, and together with their entropy empowered variant, AwU, deliver group preferences of high quality in terms of satisfying a vast majority of the group members. Indeed, their success is related neither with the number of items to be recommended nor with the groups' size, which ensures the proposed techniques' robustness.

6. Conclusions and future work

In a Group Recommender System (GRS), it is of crucial importance to aggregate preferences of the individuals in a group correctly to

identify the group preferences, as the GRS bases its recommendations to the group solely on the group preferences. The task of aggregation may be more complicated than one would expect. The groups that a GRS deals with are not often well-established, meaning that the groups may consist of users with diverse tastes. This diversity requires developing alternatives to standard approaches.

In this study, we propose to employ multiple aggregation techniques to tackle the aggregation problem in multiple dimensions. Specifically, we use Additive Utilitarian (AU) and Approval Voting (AV) in a combination to provide more chances for items on which group members reached a consensus and for highly-rated items. To this end, we offer two aggregation techniques of hybridized type: AU_{AV} and AV_{AU}. Here, AU_{AV} values AU as the decisive factor on the final group preferences, while AV_{AU} puts more emphasis on AV.

The experiments performed on three benchmark datasets with different sizes demonstrated that both AV_{AU} and AU_{AV} outperform all baseline and benchmark aggregation techniques in terms of suitability of items recommended to groups. This superiority is achieved irrespective of the size of the groups and the recommendation list's length, as proved by the statistical significance analyses. In comparing the two, the empirical results suggest that the group preferences produced by AU_{AV} fit better than those produced by AV_{AU}. Such an outcome indicates that items forming a consensus in a group are more precious than those highly rated in terms of the performance of GRSs.

Also, we suggest using rating distributions of the items to quantify the degree of consensus they provide. For this purpose, we proposed to utilize Shannon's entropy (entropy in brief) for the first time in the literature. Entropy offers two advantages over the standard deviation when employed for analyzing the dispersion of the rating distributions: (i) entropy is amenable to multiple peaks in a rating distribution (i.e., multimodal distribution), (ii) entropy is more suitable for ratings in discrete type which is often the case in GRSs.

We strengthened AU_{AV} with entropy calculation in a way that AU_{AV} ignores items with high entropy values assuming that these items do not maintain a consensus among group members. We name this technique as AwU, which stands for *Agreement without Uncertainty*. The empirical outcomes indicate that AwU is superior even to AU_{AV} based on the *n*DCG results. Practically, this means that compared with off-the-shelf aggregation techniques and the *hybridized* aggregation techniques we offer in this study, group referrals provided by AwU are more suitable for group members. Indeed, the need for employing AwU grows as the groups get more substantial. As a result, considering the degree of uncertainty of items using entropy in aggregating individual preferences contributes to providing high-quality group recommendations as the groups get crowded.

Although the performance of the proposed aggregation techniques *hybridized* and AwU is found sufficient based on the outcomes of the performed a broad set of experiments, these techniques produce group recommendations by considering only group members' experience about the domain of interest. However, social relationships among individuals play a vital role in the group's decision-making process and need to be considered during the aggregation process. Therefore, future research may include strengthening our ultimate technique AwU by incorporating social factors such as similarity and trust among group members to the aggregation process. Moreover, other baseline techniques with various formulations can be hybridized for estimating group ratings in the AwU technique. Also, rather than using k-means clustering, user groups can also be identified by other prominent clustering methods such as bisecting k-means and k-medoids.

CRediT authorship contribution statement

Emre Yalcin: Conceptualization, Methodology, Software, Investigation, Writing - original draft. **Firat Ismailoglu:** Validation, Investigation, Writing - original draft. **Alper Bilge:** Conceptualization, Methodology, Validation, Writing - review & editing, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, *17*, 734–749. http://dx.doi.org/10.1109/TKDE.2005.99.

Agarwal, A., Chakraborty, M., & Chowdary, C. R. (2017). Does order matter? effect of order in group recommendation. *Expert Systems with Applications*, *82*, 115–127. http://dx.doi.org/10.1016/j.eswa.2017.03.069.

Ahmad, H. S., Nurjanah, D., & Rismala, R. (2017). A combination of individual model on memory-based group recommender system to the books domain. In *2017 5th international conference on information and communication technology (ICoIC7)* (pp. 1–6). http://dx.doi.org/10.1109/ICoICT.2017.8074655.

Álvarez Márquez, J. O., & Ziegler, J. (2016). Hootle+: A group recommender system supporting preference negotiation. In T. Yuizono, H. Ogata, U. Hoppe, & J. Vassileva (Eds.), *Collaboration and technology* (pp. 151–166). Cham: Springer International Publishing.

Ardissono, L., Goy, A., Petrone, G., Segnan, M., & Torasso, P. (2003). Intrigue: Personalized recommendation of tourist attractions for desktop and hand held devices. *Applied Artificial Intelligence*, *17*, 687–714. http://dx.doi.org/10.1080/713827254.

Baltrunas, L., Makcinskas, T., & Ricci, F. (2010). Group recommendations with rank aggregation and collaborative filtering. In *Proceedings of the fourth ACM conference on recommender systems* (pp. 119–126). New York, NY, USA: ACM, http://dx.doi.org/10.1145/1864708.1864733.

Barzegar Nozari, R., & Koohi, H. (2020). A novel group recommender system based on members' influence and leader impact. *Knowledge-Based Systems*, *205*, Article 106296. http://dx.doi.org/10.1016/j.knosys.2020.106296.

Basu Roy, S., Thirumuruganathan, S., Amer-Yahia, S., Das, G., & Yu, C. (2014). Exploiting group recommendation functions for flexible preferences. In *2014 IEEE 30th international conference on data engineering* (pp. 412–423). http://dx.doi.org/10.1109/ICDE.2014.6816669.

Bedi, P., Agarwal, S. K., Jindal, V., & Richa (2014). Marst: Multi-agent recommender system for e-tourism using reputation based collaborative filtering. In A. Madaan, S. Kikuchi, & S. Bhalla (Eds.), *Databases in networked information systems* (pp. 189–201). Cham: Springer International Publishing.

Bekkerman, P., Kraus, S., & Ricci, F. (2006). 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).

Bilge, A., & Polat, H. (2013). A scalable privacy-preserving recommendation scheme via bisecting k-means clustering. *Information Processing & Management*, *49*, 912–927. http://dx.doi.org/10.1016/j.ipm.2013.02.004.

Bobadilla, J., Ortega, F., Hernando, A., & GutiéRrez, A. (2013). Recommender systems survey. *Knowledge-Based Systems*, *46*, 109–132. http://dx.doi.org/10.1016/j.knosys.2013.03.012.

Bobadilla, J., Serradilla, F., & Bernal, J. (2010). A new collaborative filtering metric that improves the behavior of recommender systems. *Knowledge-Based Systems*, *23*, 520–528. http://dx.doi.org/10.1016/j.knosys.2010.03.009.

Boratto, L., & Carta, S. (2014). Modeling the preferences of a group of users detected by clustering: A group recommendation case-study. In *Proceedings of the 4th international conference on web intelligence, mining and semantics (WIMS14)*. New York, NY, USA: Association for Computing Machinery, http://dx.doi.org/10.1145/2611040.2611073.

Boratto, L., Carta, S., & Fenu, G. (2016). Discovery and representation of the preferences of automatically detected groups: Exploiting the link between group modeling and clustering. *Future Generation Computer Systems*, *64*, 165–174. http://dx.doi.org/10.1016/j.future.2015.10.007.

Burke, R. (2017). Multisided fairness for recommendation. ArXiv abs/1707.00093.

Cantador, I., & Castells, P. (2011). Extracting multilayered communities of interest from semantic user profiles: Application to group modeling and hybrid recommendations. *Computers in Human Behavior*, *27*, 1321–1336. http://dx.doi.org/10.1016/j.chb.2010.07.027.

Castro, J., Yera, R., & Martínez, L. (2018). A fuzzy approach for natural noise management in group recommender systems. *Expert Systems with Applications*, *94*, 237–249. http://dx.doi.org/10.1016/j.eswa.2017.10.060.

Chao, D. L., Balthrop, J., & Forrest, S. (2005). Adaptive radio: Achieving consensus using negative preferences. In *Proceedings of the 2005 international ACM SIGGROUP conference on supporting group work* (pp. 120–123). New York, NY, USA: ACM, http://dx.doi.org/10.1145/1099203.1099224.

Christensen, I. A., & Schiaffino, S. (2011). Entertainment recommender systems for group of users. *Expert Systems with Applications*, *38*, 14127 – 14135. http://dx.doi.org/10.1016/j.eswa.2011.04.221.

Crossen, A., Budzik, J., & Hammond, K. J. (2002). Flytrap: Intelligent group music recommendation. In *Proceedings of the 7th international conference on intelligent user interfaces* (pp. 184–185). New York, NY, USA: ACM, http://dx.doi.org/10.1145/502716.502748.

Felfernig, A., Boratto, L., Stettinger, M., & Tkalčič, M. (2018). *Evaluating group recommender systems* (pp. 59–71). Cham: Springer International Publishing, http://dx.doi.org/10.1007/978-3-319-75067-5_3.

Goren-Bar, D., & Glinansky, O. (2004). Fit-recommending tv programs to family members. *Computers & Graphics*, *28*, 149 – 156. http://dx.doi.org/10.1016/j.cag.2003.12.003.

Herlocker, J. L., Konstan, J. A., Borchers, A., & Riedl, J. (1999). An algorithmic framework for performing collaborative filtering. In *Proceedings of the 22nd annual international ACM SIGIR conference on research and development in information retrieval* (pp. 230–237). New York, NY, USA: ACM, http://dx.doi.org/10.1145/312624.312682.

Jameson, A. (2004). More than the sum of its members: Challenges for group recommender systems. In *Proceedings of the working conference on advanced visual interfaces* (pp. 48–54). New York, NY, USA: Association for Computing Machinery, http://dx.doi.org/10.1145/989863.989869.

Ji, K., Chen, Z., Sun, R., Ma, K., Yuan, Z., & Xu, G. (2018). Gist: A generative model with individual and subgroup-based topics for group recommendation. *Expert Systems with Applications*, *94*, 81 – 93. http://dx.doi.org/10.1016/j.eswa.2017.10.037.

Kaleli, C. (2014). An entropy-based neighbor selection approach for collaborative filtering. *Knowledge-Based Systems*, *56*, 273 – 280. http://dx.doi.org/10.1016/j.knosys.2013.11.020.

Kaššák, O., Kompan, M., & Bieliková, M. (2016). Personalized hybrid recommendation for group of users: Top-n multimedia recommender. *Information Processing & Management*, *52*, 459 – 477. http://dx.doi.org/10.1016/j.ipm.2015.10.001.

Khazaei, E., & Alimohammadi, A. (2018). An automatic user grouping model for a group recommender system in location-based social networks. In *ISPRS international journal of geo-information* (p. 67). http://dx.doi.org/10.3390/ijgi7020067.

Lieberman, H., van Dyke, N., & Vivacqua, A. (1999). Let's browse: a collaborative browsing agent. *Knowledge-Based Systems*, *12*, 427–431. http://dx.doi.org/10.1016/S0950-7051(99)00036-2.

Liu, Y., Wang, B., Wu, B., Zeng, X., Shi, J., & Zhang, Y. (2016). Cogrec: A community-oriented group recommendation framework. In *Social computing* (pp. 258–271). Singapore: Springer Singapore, http://dx.doi.org/10.1007/978-981-10-2053-7_24.

Lu, J., Wu, D., Mao, M., Wang, W., & Zhang, G. (2015). Recommender system application developments. *Decision Support Systems*, *74*, 12–32. http://dx.doi.org/10.1016/j.dss.2015.03.008.

Mahyar, H., Ghalebi, K. E., Morshedi, S. M., Khalili, S., Grosu, R., & Movaghar, A. (2017). Centrality-based group formation in group recommender systems. In *Proceedings of the 26th international conference on world wide web companion, international world wide web conferences steering committee* (pp. 1187–1196).

Masthoff, J. (2015). Group recommender systems: Aggregation. In *Satisfaction and group attributes* (pp. 743–776). Boston, MA: Springer US, http://dx.doi.org/10.1007/978-1-4899-7637-6_22.

McCarthy, J. F. (2002). Pocket restaurantfinder: A situated recommender system for groups. In *Workshop on mobile ad-hoc communication at the 2002 ACM conference on human factors in computer systems*.

McCarthy, J. E., & Anagnost, T. D. (2000). Musicfx: An arbiter of group preferences for computer supported collaborative workouts. In *Proceedings of the 2000 ACM conference on computer supported cooperative work* (p. 348). New York, NY, USA: Association for Computing Machinery, http://dx.doi.org/10.1145/358916.361976.

McCarthy, K., Salamó, L., McGinty, L., Smyth, B., & Nixon, P. (2006). Cats: A synchronous approach to collaborative group recommendation. In *FLAIRS 2006 - Proceedings of the nineteenth international florida artificial intelligence research society conference* (pp. 86–91). AAAI Press.

Nguyen, T. N., & Ricci, F. (2018). A chat-based group recommender system for tourism. *Information Technology & Tourism*, *18*, 5–28. http://dx.doi.org/10.1007/s40558-017-0099-y.

Ntoutsi, I., Stefanidis, K., Norvag, K., & Kriegel, H. P. (2012). Grecs: A group recommendation system based on user clustering. In S. g. Lee, Z. Peng, X. Zhou, Y. S. Moon, R. Unland, & J. Yoo (Eds.), *Database systems for advanced applications* (pp. 299–303). Springer Berlin Heidelberg, http://dx.doi.org/10.1007/978-3-642-29035-0_25.

O'Connor, M., Cosley, D., Konstan, J. A., & Riedl, J. (2001). *PolyLens: a recommender system for groups of users* (pp. 199–218). Dordrecht: Springer Netherlands, http://dx.doi.org/10.1007/0-306-48019-0_11.

Quijano-Sanchez, L., Recio-Garcia, J. A., & Diaz-Agudo, B. (2011). Happymovie: A facebook application for recommending movies to groups. In *2011 IEEE 23rd international conference on tools with artificial intelligence* (pp. 239–244). http://dx.doi.org/10.1109/ICTAI.2011.44.

Ricci, F., Rokach, L., & Shapira, B. (2011). *Introduction to recommender systems handbook* (pp. 1–35). Boston, MA: Springer US.

Sacharidis, D. (2019). Top-n group recommendations with fairness. In *Proceedings of the 34th ACM/SIGAPP symposium on applied computing* (pp. 1663–1670). New York, NY, USA: Association for Computing Machinery, http://dx.doi.org/10.1145/3297280.3297442.

Salehi-Abari, A., & Boutilier, C. (2015). Preference-oriented social networks: Group recommendation and inference. In *Proceedings of the 9th ACM conference on recommender systems* (pp. 35–42). New York, NY, USA: Association for Computing Machinery, http://dx.doi.org/10.1145/2792838.2800190.

Seo, Y. D., Kim, Y. G., Lee, E., Seol, K. S., & Baik, D. K. (2018). An enhanced aggregation method considering deviations for a group recommendation. *Expert Systems with Applications*, *93*, 299 – 312. http://dx.doi.org/10.1016/j.eswa.2017.10.027.

Serbos, D., Qi, S., Mamoulis, N., Pitoura, E., & Tsaparas, P. (2017). Fairness in package-to-group recommendations. In *Proceedings of the 26th international conference on world wide web*. (pp. 371–379).

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, *27*, 379–423.

Villavicencio, C., Schiaffino, S., Andres Diaz-Pace, J., & Monteserin, A. (2019). Group recommender systems: A multi-agent solution. *Knowledge-Based Systems*, *164*, 436 – 458. http://dx.doi.org/10.1016/j.knosys.2018.11.013.

Wang, X., Liu, Y., Lu, J., Xiong, F., & Zhang, G. (2019). Trugrc: Trust-aware group recommendation with virtual coordinators. *Future Generation Computer Systems*, *94*, 224–236. http://dx.doi.org/10.1016/j.future.2018.11.030.

Yalcin, E., Bilge, A., & Yuksek, A. G. (2019). An empirical evaluation of aggregation techniques used in group recommender systems. In *Proceedings of the 8th international conference on advanced technologies (ICAT 2019)* (pp. 186–191).

Yargic, A., & Bilge, A. (2019). Privacy-preserving multi-criteria collaborative filtering. *Information Processing & Management*, *56*, 994 – 1009. http://dx.doi.org/10.1016/j.ipm.2019.02.009.

Yera, R., & Martínez, L. (2017). Fuzzy tools in recommender systems: A survey. *International Journal of Computational Intelligence Systems*, *10*, 776–803. http://dx.doi.org/10.2991/ijcis.2017.10.1.52.

Zhiwen, Y., Xingshe, Z., & Daqing, Z. (2005). An adaptive in-vehicle multimedia recommender for group users. In *2005 IEEE 61st vehicular technology conference* (pp. 2800–2804). IEEE, http://dx.doi.org/10.1109/vetecs.2005.1543857.