International Journal of Approximate Reasoning 150 (2022) 273–296

Contents lists available at ScienceDirect

International Journal of Approximate Reasoning

www.elsevier.com/locate/ijar

A fuzzy content-based group recommender system with dynamic selection of the aggregation functions

Raciel Yera [a,c], Ahmad A. Alzahrani [b], Luis Martínez [a,*]

[a] *Computer Science Department, University of Jaén, Jaén, Spain*
[b] *Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia*
[c] *University of Ciego de Ávila, Ciego de Ávila, Cuba*

A R T I C L E   I N F O

Article history:
Received 13 January 2022
Received in revised form 28 June 2022
Accepted 24 August 2022
Available online 7 September 2022

Keywords:
Content-based recommendation
Group recommender systems
Dynamic aggregation selection
Machine learning

A B S T R A C T

Recommender systems are currently software tools that are focused on providing users with the best choices in an overloaded search space of possible options. Hence, group recommender systems have recently become an important trend in recommendation, because they aim at recommending a special type of items so-called social items, that tend to be consumed in groups such as TV programs, travel packages, etc. Among the different types of algorithms applied for group recommender systems, this paper is focused on content-based group recommender systems, as a novel group recommendation paradigm that exploits item features in the recommendation generation process. Specifically, our goal is to introduce a new content-based group recommendation approach, based on the recommendation aggregation paradigm whose main novelty is the development of a dynamic selection process of the aggregation scheme. Such an approach is centered on the identification of group's characteristics that are matching with the most appropriate function to use in the individual recommendation aggregation step. To perform such a matching, it is proposed a fuzzy decision tree induction process. The experimental evaluation shows that this scheme improves the recommendation performance of previous content-based group recommendation approaches, as well as it serves a starting point for further research based on this dynamic selection paradigm.

1. Introduction

Recommender systems (RSs) are currently important tools in online scenarios, focused on providing suggestions to users with items that best fit their preferences and needs, in an overloaded product search space of possible options [1,23,52]. Considering their working principles, RSs have been widely used in several and diverse domains, such as e-commerce [4], e-learning [53], e-health [50], or e-tourism [35].

Two main directions have driven the development of RSs. At first, the content-based recommendation paradigm [33] is focused on recommending to the active user items which have similar characteristics to other items previously preferred by the same user. This paradigm is then mainly focused on user and item profiling for reaching a more representative matching between them. On the other hand, the collaborative filtering paradigm [21] is focused on recommending to the current user items which are preferred by other users similar to the active one. Specifically, collaborative filtering methods can be classified in neighborhood-based or model-based methods [1].

* Corresponding author.
E-mail addresses: ryera@ujaen.es (R. Yera), aalzahrani8@kau.edu.sa (A.A. Alzahrani), martin@ujaen.es (L. Martínez).

https://doi.org/10.1016/j.ijar.2022.08.015

These recommendation paradigms have been usually used to build RSs for providing suggestions to individual users. However, in the last few years, several kinds of items that tend to be consumed by groups, have appeared in recommendation contexts [9,16]. As example of such items, can be cited movies and touristic routes [17,42]. The recommendation of such items requires an additional effort in relation to individual recommendation, because it should manage the preferences both at the individual and the group level. Such a requirement has made the development of group recommender systems (GRSs) as an independent research branch in RS field [19].

Basically, GRSs are centered on aggregating the information associated to the group members [19]. Such an aggregation can be done by recommendation aggregation, in which first it is computed individual recommendations for every group member, and then such recommendations are combined through a recommendation aggregation approach. Alternatively, a preference aggregation can also be used, where it is created a pseudo-user that globally represents the preferences of the group and such a pseudo-user profile is used for computing the group recommendation.

Using previous schemes, several research works have been focused on proposing new GRSs approaches, having as common characteristic the use of a collaborative filtering approach as a core of the recommendation method [10,13,44,47,15,49], motivated by the advantages related to collaborative filtering, in relation to the ability of generating recommendations using only rating values. However, a well-documented shortcoming of collaborative filtering is the poor performance in highly sparse recommendation scenarios including cold-start [45], considering that it depends on the presence on items co-evaluated by several users for a proper performance. In contrast, content-based group recommender systems (CB-GRSs, including the proposal developed at the current work), usually obtain a greater success in such scenario, considering that it is different from collaborative filtering because they only depend on the current user preference data and the attributes' information associated to the available items.

Nevertheless, the recent research literature reflects too few efforts for boosting the use of CB-GRS. De Pessemier et al. [19], in one of the firstly documented surveys on GRS, slightly mentioned and evaluated an alternative for group recommendation supported by item features for predicting user preferences. However, this work is mainly focused on comparing social-choice based group recommendation algorithms, and therefore do not perform an in-depth analysis of the specific content-based scenario. Furthermore, the architecture of a content-based recommendation algorithm has been screening by Felfernig et al. [22], but again such presentation lacks of a detailed analysis of each component and experimentation. Recently, Pérez-Almaguer et al. [41] have discussed three basic design alternatives for building CB-GRSs, that consider the aggregation paradigm (preferences-driven or recommendations-driven), the way of aggregating recommendations (ranking-based or similarity-based), the social choice-based schemes to perform aggregation (e.g. average, least misery, most pleasure), and other relevant design decisions. That work also explores the possible hybridization between some of the presented schemes. In addition, other research works have also exploited the content-based dimension in group recommendation, with a greater or lesser extent [39,31].

This lack of works in CB-GRS in spite of its promising but not achieved great performance, evidences the necessity of developing more sophisticate approaches for CB-GRS, in order to obtain a better recommendation performance. This paper aims at proposing a new CB-GRS scheme centered on taking benefits of the nature of the data, for improving the recommendation performance. Particularly, we are interested in proposing a *dynamic* aggregation process, in the recommendation aggregation step in CB-GRS, that will choose an appropriate aggregation function according to the group's characteristics.

According to the Cambridge Dictionary, the word *dynamic*[1] is related to a *continuous changing or developing*. In this case, we have used the *dynamic* tag in our proposal, to make reference to the nature of the function used to aggregate the preference of the member of the groups. In this way the selected function, used with this goal, will be continuously changing based on the characteristics of the current group.

Considering that the aggregation process can produce loss of information [19], the choice of the appropriate aggregation function can lead to decrease the information loss and consequently improving the recommendation performance.

In this direction, this research is driven by the following *problem definition*: Given a set G of groups of users and a specific group c, find the most appropriate function f to aggregate individual preferences in the recommendation aggregation step inside the group, which leads to the better recommendation performance in a CB-GRS framework.

Focused on this research problem, the main contributions of our proposal are:

- A new CB-GRS approach that introduces a component for the dynamic selection of the recommendation aggregation function for composing group recommendations. This component will receive the current group features, and returns the most suitable aggregation function for such a group characterization.
- The inner modeling of the dynamic selection component, done as a supervised classification scenario using classification rules. Such classification rules are obtained by a fuzzy classification tree using the ID3 algorithm [46,30]. From a machine learning perspective, our proposal can be seen as a meta-learning approach.
- An exploratory study of group's features in CB-GRS, such as amount of ratings, minimum user's correlation, or amount of co-rated items, that could contribute to a better characterization of groups and therefore improving their recommendations.
- An experimental study in order to evaluate the proposal, in contrast to previous baselines.

[1] https://dictionary.cambridge.org/dictionary/english/dynamic.

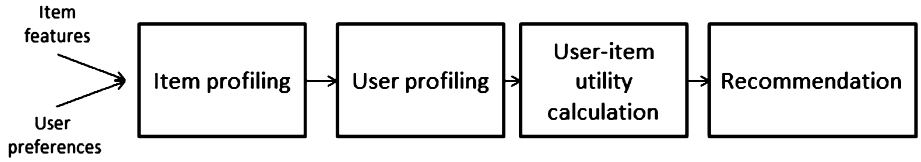

Fig. 1. General scheme of content-based recommendation.

The paper is structured as follows. Section 2 presents the necessary background for the proposal presentation, including content-based recommendation, group recommendation, and previous works in content-based group recommendation. Section 3 introduces the new CB-GRS proposal, supported by the dynamic selection of the aggregation function. Subsequently, a case study is also presented to show how the proposal works (Section 4). Furthermore, Section 5 is focused on evaluating the proposal, discussing the main findings and pointing out future works. Section 6 concludes the paper.

2. Background

This section is focused on presenting some concepts that are necessary for the later proposal presentation. This content involves fundamentals on content-based recommendation and group recommender systems. At last, a related works section on CB-GRS will be briefly introduced.

2.1. Content-based recommendation

Since 90s, content-based recommender systems have been widely used as one of the most popular recommendation approaches [1]. Here the recommendation process is composed of the following steps (see Fig. 1):

- The first step in content-based recommendation comprises the construction of an item profile $Content(i)$, that is represented by a set of features that can be explicitly or implicitly associated to the item i. A key example of explicit features, can be genre, director, country or year, in a movie recommendation scenario. On the other hand, implicit features are usually identified with techniques such as the latent semantic analysis (LSA), in domains such as news or question-answering item recommendation [12]. In such an scenario, the TF-IDF approach is usually used for managing the free-text items. Specifically, such a text is converted into structured data stemming words, and after that, a vector of weights of each term is generated, according to the TF-IDF scheme. Finally, techniques such as LSA are applied for a most precise item representation [33].
- The second step of any content-based recommender systems is the user profiling. Usually, this profile $ContentBased Profile(u)$ is built by fusing the profiles of all the items preferred by the active user u. Several strategies have been proposed for performing such a fusion, including the use of computational intelligence techniques [1,33].
- Both previous profiles are used to calculate the utility of item i for user u. Such utility $v(u,i) = score(ContentBased Profile(u), Content(i))$ is usually represented with a similarity measure such as cosine [1]. In this way, cosine metric has been used for comparing profiles linked to the vector space model (e.g. keywords-based profiles, term-based profiles), usually related to content-based recommendations [1,37,24]. More complex approaches focused on semantic similarity have been also incorporated [40], however their application depends on further knowledge sources (e.g. ontologies, linked open data cloud, etc), which incorporation goes beyond the current proposal of this work that is the screening of a new CB-GRS approach.
- Finally, the top-n items with the higher utility, will be suggested for the active user u.

Most of research works on recommender systems have been focused on individual users. However, since the last decade there is an increasing in the interest over group recommender systems (GRSs) as a novel recommendation scenario. The next section is focused on briefly presenting the fundamentals of GRSs.

2.2. Group recommender systems

The appearing of GRSs has been coupled with the need of recommending some items that are usually consumed in groups, such as movies, touristic routes, or TV programs [14,19]. In such cases, it is necessary to recommend items that maximize the overall satisfaction of the group. With this goal in mind, GRSs extend individual RS, by taking into account the aggregation of information related to each individual group member.

The literature identifies two main paradigms in group recommendation, based on the nature of the information aggregation approach [19]:

- *Rating aggregation:* This approach combines the preferences of the individual users, to build a pseudo-user profile that is later used as a typical user profile to receive recommendations that in this case are delivered to the group (Fig. 2).

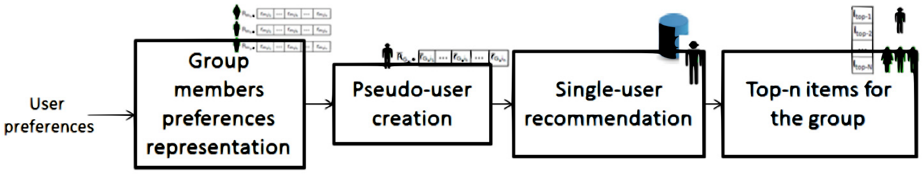

Fig. 2. Group recommendation based on rating aggregation.

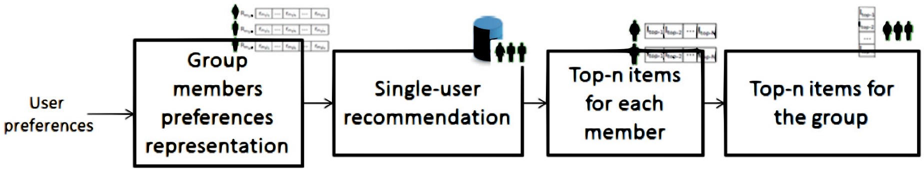

Fig. 3. Group recommendation based on recommendation aggregation.

Table 1
Relevant notation.

Term	Meaning
u	User
i	Item
G	Group
f_k^i	Value of the feature k for item i
f_k^u	Value of the feature k for user u
f_k^G	Value of the feature k for group G
$V^k = \{v_1^k, v_2^k, v_3^k, ..., v_p^k\}$	Possible values of the feature k in the item profile, for multivalued features
v^{ki}	Value of the feature k in the item i, for multivalued features
v_p^{ku}	pth value of the feature k for the user u, for multivalued features
top_u	List of top n recommendations for user u
S_{ui}	Matching value between user u and item i
S_i^G	Matching value between group G and item i
I_G	Top k items recommended to the group G

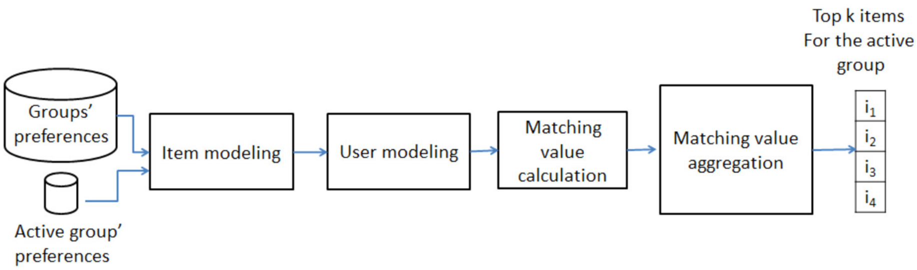

Fig. 4. CB-GRS based on recommendation aggregation and user-item matching values.

- *Recommendation aggregation:* This approach at first generates individual recommendations for each member of the group. Such individual recommendations are then aggregated to compose the final recommendation list for the group (Fig. 3).

The following subsection will present in further detail a recent GRS model built over the content-based recommendation paradigm [41], which will be used as starting point for the proposal developed at the current paper.

2.2.1. CB-GRS based on recommendation aggregation and user-item matching values

This section describes in further detail, the CB-GRS approach based on recommendation aggregation and user-item matching values, initially presented at Pérez-Almaguer et al. [41], where it was evidenced that it is able to outperform other GRS models including collaborative filtering-based. Table 1 presents the notation used across this section. This basic content-based group recommendation model (Fig. 4) is composed of four phases: 1) Item modeling, 2) User modeling, 3) User-item matching value calculation, and 4) Matching value aggregation for obtaining the top k items for the group.

Item modeling: In a similar way to the typical content-based recommendation, this step is focused on representing the items to be recommended, through the modeling of a feature vector (Fig. 4). Considering that the information associated to items can be represented through different formats, here it will be considered two ways for modeling items:

1. A basic approach that considers a binary profile that contains 1 whether the item has the corresponding feature, and 0 whether the item does not contain it. Formally, items are represented as the vector $i = (f_1^i, f_2^i, ..., f_m^i)$, where $f_k^i = 1$ whether the feature k is associated to the item i, and $f_k^i = 0$ otherwise.
2. A more sophisticated approach that considers multivalued features [11]. In this case, items are also represented as the vector $i = (f_1^i, f_2^i, ..., f_m^i)$, but here f_k^i is associated to nominal or numeric values, in a domain associated to the feature k [11].

User modeling: In a similar way to items, here it will be considered two approaches for user modeling.

1. An approach based on TF-IDF [2], considering the preferred items. This approach assumes a binary item profile, and here users are represented through a vector $u = (f_1^u, f_2^u, ..., f_m^u)$. f_k^u is defined as:

$$f_k^u = FF(u,k) * IUF(k) \tag{1}$$

where $FF(u,k)$ is calculated as the number of items i preferred by the user u, having $f_k^i = 1$ where i is any item profile built in the previous item modeling phase. On the other hand, $IUF(k) = log\frac{|U|}{UF(k)}$, being $UF(k)$ the number of users that have preferred any item that has the feature k, and $|U|$ the total number of users.
2. An approach that assumes the presence of multivalued features [11], having the presence of nominal or numeric values in the item features. In this scenario, it is necessary a new formulation of the user profile (Eq. (2)).

$$f_k^u = \begin{cases} \{(v_1^k, fr_{v_1^{ku}}), (v_2^k, fr_{v_2^{ku}}), (v_3^k, fr_{v_3^{ku}}), ..., (v_p^k, fr_{v_p^{ku}})\}\ , & \text{if } k \text{ is qualitative} \\ \text{average}(f_k^i) \text{ for each item i preferred by } u, & \text{if } k \text{ is quantitative} \end{cases} \tag{2}$$

For items associated to qualitative features, f_k^u is formalized as a set of pairs (value, frequency) composed of each one of the possible values v_p^k of the feature k, and the frequency $fr_{v_p^{ku}}$ of such value at the feature k in all the items preferred by the user u.
In the case of quantitative features in the items, f_k^u will be the average of all the values associated to the feature, for all items preferred by the user u.

User-item matching value calculation: Subsequently, the current content-based GRS requires the calculation of the matching degree between the corresponding user and item profile [11]. Depending on the presence of a binary user profile or multivalued features, a different approach will be used:

1. For the binary item profiles, it will be directly used the cosine similarity function between the user and item profiles u and i (Eq. (3)) [1], as the reference metric for content-based recommendation (see Section 2.1).

$$S_{ui} = \frac{\sum_{u,i} f_k^u * f_k^i}{\sqrt{(f_k^{u^i})^2}\sqrt{(f_k^i)^2}} \tag{3}$$

2. In the case of the items with multivalued features, at first it is necessary to define the matching value between users and items, but in the context of a specific feature k (Eq. (4)). For qualitative features, this value is calculated as fr_{v^k}, being v the associated key in the list of pairs at f_k^u, as well as the value at f_k^i. For quantitative features, this value is calculated as the inverse of the difference between f_k^u and f_k^i.

$$S_{ui}^k = \begin{cases} fr_{v^k}\ , & \text{for } k \text{ qualitative} \\ \frac{1}{|f_k^u - f_k^i|}\ , & \text{for } k \text{ quantitative} \end{cases} \tag{4}$$

Such matching values are normalized independently for the qualitative and quantitative scenarios. The matching values are then denoted as S_{ui}^{k*}.
At last, the overall matching value between the user u and item i is calculated as the average matching value of all the features (Eq. (5)), being K the set of item features:

$$S_{ui} = \frac{\sum_{k \in K} S_{ui}^{k*}}{|K|} \tag{5}$$

Matching value aggregation for obtaining the top k items for the group: In the next step, the method depends on an aggregation function to obtain the matching values associated to all the group's members, and each item in the dataset.

The popular Average and Minimum aggregation functions will be considered in this scenario [19], which have been reported in previous research in GRS as the aggregation measures that lead to a better performance regarding typical alternatives [13,9], and specifically focused on CB-GRS [41]. Furthermore, more sophisticated aggregation measures such as those recently discussed by Yalcin et al. [49] will be explored in the next future. However, they are out of the scope of this work.

The Average and Minimum aggregation functions are formalized as follows:

1. Average: It calculates the average matching value for all the users in the group, being n the number of users.

$$S_i^G = \frac{\sum_{u \in G} S_{ui}}{n} \tag{6}$$

2. Minimum: It assumes a fairness-aware approach, and considers the lowest matching value in the group, as the aggregated matching value.

$$S_i^G = Min_u \;\; S_{ui}, \; \forall u \in G \tag{7}$$

At last, the available items are sorted in descending order according to their aggregated matching values, retrieving the top N items I_G as the recommendation list for the active group G.

Beyond this recently proposed CB-GRS approach, the next section will discuss the previous works developed by the research community, focused in this area with a larger or lesser extent.

2.3. Related works

Several works have been focused on CB-GRS in recent years, with a larger or lesser extent.

An early work in this direction was presented by Pera & Ng [39], focused on proposing a GRS for movies that uses content similarity and popularity as active information for the recommendation generation. The group is modeled through an aggregation component that merges individual group's member profiles. Such profiles are represented through the tags associated to the movies by the members themselves. A movie is then regarded as a candidate movie for recommendation, if each of the personal tags of the group, are highly similar to a tag directly associated to the movie. Offline experiments were developed to evaluate this approach.

Afterwards, De Pessemier et al. [19], in a well-recognized survey on GRS, make reference to a basic scheme of CB-GRS, as one of the alternatives for the individual recommendation step in the GRS framework. Here an experimental procedure focused on comparing several GRS architectures, does not reflect any relevant finding or further analysis on the use of the content-based approach, in contrast to other collaborative filtering-based methods.

Kaššák et al. [31] more recently provide a new GRS method that is based on the integration of an individual collaborative filtering and a content-based method. Here, the recommendation list initially delivered to the group' members are aggregated for combining the output of both recommendation schemes. The authors also developed offline experiments using the Movielens dataset, as well as online experiments with real deployed systems.

A relevant book on group recommendation, presented by Felfernig et al. [22], introduces a chapter section characterizing the architecture of a content-based recommendation algorithm, including stages such as content-based filtering per user, and aggregation of user-item similarities. Furthermore, it also exposes some guidelines for user and item profiling. However, the topic related to content-based at such book, lacks of a enough detailed analysis of each component and experimentation.

Recently, Pérez-Almaguer et al. [41] have also discussed three basic design alternatives for building CB-GRS. Such alternatives are: 1) the content-based GRS based on recommendation aggregation using individual rankings, 2) the content-based GRS based on recommendation aggregation and user-item matching values, and 3) the content-based GRS based on the aggregation of the user profiles. They also explore the possible hybridization between some of the presented schemes. The experimental framework presented, includes a component for evaluating the proposals in a cold-start scenario, showing that the proposals are able to outperforms a collaborative filtering approach in this context.

Other research works in the last few years have also incorporated the content-based dimension in their proposal [18,34,56]. However, these works do not consider the individual analysis and evaluation of the content-based component. Therefore, they are not further described in this analysis.

Taking as base the CB-GRS model developed by Pérez-Almaguer et al. [41], the aim of this paper is focused on introducing a novel paradigm that has not been considered previously in CB-GRS, and either in any other general GRS context as far as we know. Such paradigm is the dynamic selection of the recommendation aggregation function for composing group recommendations, by receiving as input some current group attributes, and retrieving the best aggregation function to use.

3. Content-based group recommendation supported by dynamic selection of the aggregation function

This section introduces a novel procedure for building CB-GRSs based on *the user-item matching value aggregation step for calculating the global preferences of the group*. While the previous reported works statically use a predefined aggregation function for this stage, here it is proposed the use of a dynamic selection process of the aggregation function to be considered in the aggregation of the group's member preferences to obtain the final recommendations (Fig. 5). This process is supported through the use of rules obtained from a fuzzy decision tree that is built over some characteristics of the group that would influence the performance of the different aggregation functions.

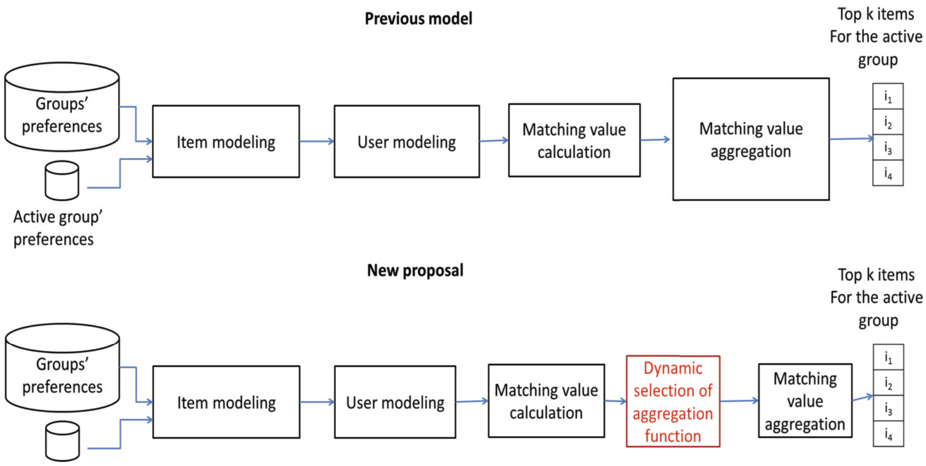

Fig. 5. Screening the novel procedure for building CB-GRSs.

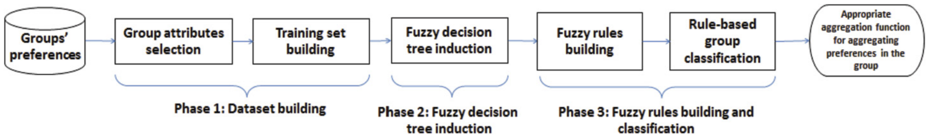

Fig. 6. The stages of the process focused on the dynamic selection of the aggregation function.

To reach this goal, this process is composed of three phases: dataset building, fuzzy decision tree induction, and fuzzy rules building and classification (Fig. 6):

1. *Dataset building:* This phase is focused on gathering all the necessary data for the subsequent phases. It includes the group attributes selection and the training set building.
2. *Fuzzy decision tree induction:* This phase is focused on building the fuzzy decision tree, taking as input the built training set represented by the groups' profile composed of the group attribute values.
3. *Fuzzy rules building and classification:* This phase is focused on using the built decision tree for obtaining fuzzy rules that will compose a rule-based classifier. Such classifier will allow, for a specific group, to determine the most appropriate aggregation function for aggregating its member preferences in the group recommendation task.

In detail, we propose the modeling of a direct matching between a set of group features and such proper aggregation function, by using a fuzzy supervised classification approach. Here, we incorporate a previously-trained supervised classifier that receives as input the attributes of the active group, and provides as output which aggregation function [19], is the most appropriated one to be used for aggregating the matching values associated to each individual of the group.

With this aim, we have selected a fuzzy ID3 classifier [46,30], which has several advantages over the other alternatives, considering that it is a white-box and therefore an understandable and interpretable model [3], and that also is able to manage uncertainty through the use of fuzzy logic which is necessary in recommender systems [52].

Fig. 7 depicts this process, for illustrating its working principle. Here it is assumed two groups' features X and Y, which are characterized by three linguistic terms low, *medium*, and *high*. The objective of the current proposal is the building of a fuzzy decision tree like the presented in the left box of Fig. 7, which leads to the classification rules presented in the right box. Based on the values of the attributes X and Y for each group, these rules make decision about which aggregation function is better for aggregating the individual preferences. In this proposal the decision is between Average or Minimum operators, considering they have had a good performance in previous CB-GRS scenarios (see Section 2.2.1).

This approach is further detailed in the coming subsections.

3.1. Dataset building

The dataset building phase comprises all the necessary data management operations as a previous step for the fuzzy decision tree induction. This phase is composed of two steps, 1) Group attributes selection, and 2)Training set building.

Group attributes selection: The first step of this phase, requires the identification of attributes for characterizing groups. While the features/attributes extraction paradigm has been widely used in several recommender systems models [32], the attributes extracted by such models usually do not have a clear semantic meaning [27]. In contrast, in our current contribution it is necessary to characterize groups through attributes with a clear semantic meaning, regarding they will be

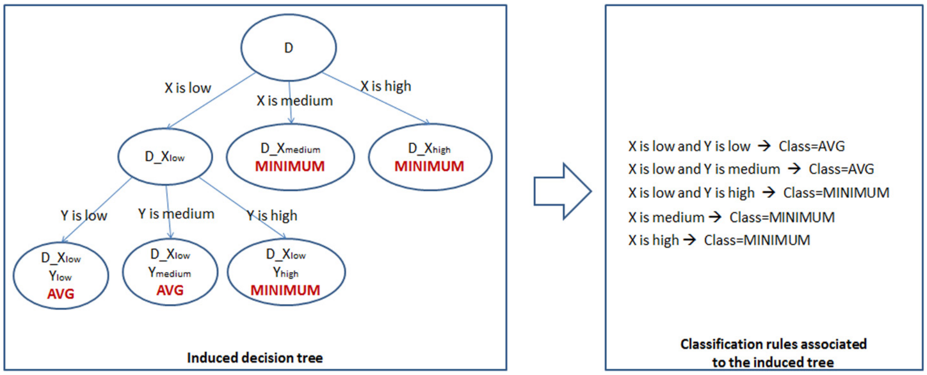

Fig. 7. Overview of the fuzzy decision tree induction and classification process, for supporting the dynamic selection of the aggregation function.

Table 2
Attributes for characterizing the groups.

Term	Meaning
M	Minimum correlation between any pair of group members
A	Amount of ratings overall, provided by all the group members
C	Amount of co-rated items by all the users in the group
AV	The rating average of the group

incorporated later in a white-box computational model that require to understand their nature (i.e. a fuzzy decision tree, see Fig. 7).

For this reason, in the current contribution we characterize groups through an attribute building process that follows the common sense and also previous criteria by other authors, for proposing attributes that characterize groups in a GRS scenario. The analysis of the research literature also identifies the development of similar attributes building procedure in previous works, such as [28] in the travel recommendation domain, [29] in job recommendation, [48] in music preference prediction, or [55] in multi-criteria recommendation.

In this way, in the current work the following group attributes are explored: 1) the minimum correlation between any pair of group members (M), 2) the amount of ratings provided by the group (A), 3) the amount of co-rated items by all the users (C), and 4) the rating average of the group (AV) (see Table 2). Previous works have suggested that such kind of information can be used for characterizing groups in a distinguishable way, and that the proper aggregation approach in this context, could depend on such features [5,19,6,13,8].

Equation (8) formalizes the minimum correlation between any pair of group members (M). In this case, for each pair of users in the group, the Pearson's correlation coefficient considering their rating values is calculated. Finally, the group is characterized as the minimum of such correlation values.

$$M(G) = min \;\; corr(u, v), \; \forall u, v \in G \tag{8}$$

For each group, the calculation of this value $M(G)$ has a computational cost $O(nlogn)$, being n the amount of ratings for the user with the larger list of ratings.

Equation (9), on the other hand, formalizes the amount of ratings elicited by the group (A), which can be obtained in a direct way. Here R_u is the set of the ratings provided by the user u.

$$A(G) = \sum_{u \in G} |R_u| \tag{9}$$

For each group, the calculation of this value $A(G)$ has a computational cost $O(n)$, being n the amount of ratings for the user with the larger list of ratings.

Subsequently, the amount of co-rated items (C) is formalized in Equation (10). In this case, this attribute is focused on the amount of items that has been evaluated by all the members of the group.

$$C(G) = |I_c|, \; where \; I_c = \{i \; : \; \forall_{u \in G} r_{ui} \in R\} \tag{10}$$

For each group, the calculation of this value $C(G)$ has a computational cost $O(nlogn)$, being n the amount of ratings for the user with the larger list of ratings. Considering the computational cost viewpoint, the reaching of $C(G)$ could be identified as a subtask of the $M(G)$ calculation.

Finally, Equation (11) formalizes the simple average (AV) of all the ratings in the group's members, as the fourth group attribute. Here, R is the set of ratings provided by all the members in the group (Equation (12)).

$$AV(G) = \frac{\sum_{r_{ui} \in R} r_{ui}}{|R|} \quad (11)$$

$$R = \cup_{u \in G} R_u \quad (12)$$

For each group, the calculation of this value $AV(G)$ has a computational cost $O(nlogn)$, being n the amount of ratings for the user with the larger list of ratings.

In addition to the low computational cost associated to the calculation of these four features, it is worthy to note that in practice, their values were instantaneously obtained for all experimental scenarios developed in this paper (Section 5).

Without loss of generality, we assume that the attribute values characterizing each group, are already normalized (into the range $[0, 1]$), facilitating the subsequent stages of the proposal.

Remark 1. For complementing the reported evidence on the suitability of these attributes, Section 5.4 presents an exploratory study for characterizing them in real RS datasets. This study concludes that they are an appropriate way for identifying different kind of groups, being such fact relevant for the personalized recommendation delivery.

Training set building: From the identified attributes for characterizing groups, the training set building process is focused on building a dataset able to execute a supervised classification task using such attributes as features, and the most appropriate aggregation function as target class. It is developed by the following procedure:

1. At first, several groups from the available data are sampled. It is expected that each sampled group is characterized by several numeric attributes, having a close relationship with the member's ratings. Therefore, initially is necessary to compute and normalize the four attribute values associated to the group. The values of such attributes M, A, C, AV, are used for characterizing groups in a supervised classification scenario deployed here.
2. Once each sampled group is characterized by their corresponding attributes values, it is necessary to find the class $C \in \{Average, Minimum\}$ associated to each group. These two classes have been considered because previous works have reported that they performed well in a CB-GRS scenario [41]. The class in this case represents the aggregation function that performs best at the matching value aggregation step for obtaining the top k items for the group, presented at Section 2.2.1. To find it, we use the following steps:
   - Execute twice the core content-based GRS already presented at Section 2.2.1 for the current group and sampled dataset, initially using the average aggregation and at second using minimum aggregation.
   - Compare both recommendation approaches according to some specific evaluation method. In this case, it will be used the precision metric which has been an appropriate metric for characterizing content-based GRS, according to the most recent literature [41]. In the future, other metrics such as NDCG, or even a multicriteria approach simultaneously considering several metrics, will be developed.
   - Finally, the class of the current group is tagged as the aggregation scheme (average or minimum) that performs best according to the evaluation metric. Future works will also consider other aggregation operators like the maximum aggregation in this context [19]. However, we discard it at this moment because it leads to poor recommendation performance in recent evaluations done by the literature [41].

Overall, this phase retrieves as output a dataset containing several group profiles, each one characterized by their associated attribute values. Furthermore, each group is also linked to a class value that represents the aggregation function that leads to the best recommendation performance (i.e. average or minimum). This dataset is used as input for the next phase of the proposal, which is the fuzzy decision tree induction.

3.2. *Fuzzy decision tree induction*

Here, it is presenting the fuzzy decision tree induction process, made in this case over the group characterization, built in the previous step.

This phase assumes D as fuzzy set, that is characterized by a membership value for each group G in the dataset. (Initially with membership 1 for all groups, at the root of the tree). Furthermore, each group is represented by four numerical values respectively for attributes $A_i \in \{M, A, C, AV\}$, and one class $C_k \in \{Average, Minimum\}$ (see Section 2.2.1 for justifying the classes selection). In addition, assumes D^{C_k} as a fuzzy subset of D, where $\mu_{D^{C_k}}(G) = \mu_D(G)$ if the class of G is C_k, and $\mu_{D^{C_k}}(G) = 0$ otherwise. Finally, $|D^{C_k}|$ is the cardinality of the fuzzy set D^{C_k}, defined as the sum of the membership value of each associated object [46].

In the current scenario, it is considered that each attribute A_i, always represented by numerical values, will be characterized by three triangular fuzzy sets low, $medium$, and $high$ (Fig. 8). More complex fuzzy representations could be also used, for modeling this membership. Therefore, each group is then characterized by the membership values for each mentioned fuzzy sets, considering each of the four attributes (Table 3).

The algorithm for constructing the fuzzy decision tree is then as follows:

1. Initially build the root node which is composed of all the data, and then is represented as a fuzzy set with all the objects having 1 as membership value.

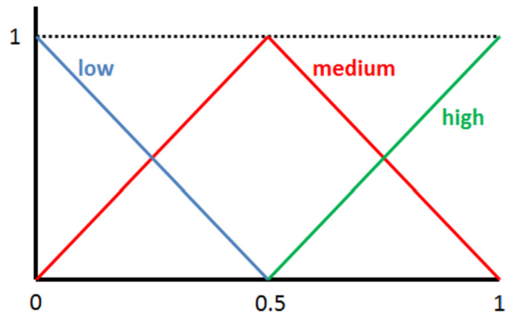

Fig. 8. Membership functions.

Table 3
Representation of each group, using the four attributes and the corresponding fuzzy sets *low*, *medium*, and *high*.

g_1	$(\mu_{M,low}(g_1), \mu_{M,medium}(g_1), \mu_{M,high}(g_1), \mu_{A,low}(g_1), \mu_{A,medium}(g_1), \mu_{A,high}(g_1),$ $\mu_{C,low}(g_1), \mu_{C,medium}(g_1), \mu_{C,high}(g_1), \mu_{AV,low}(g_1), \mu_{AV,medium}(g_1), \mu_{AV,high}(g_1))$
g_2	$(\mu_{M,low}(g_2), \mu_{M,medium}(g_2), \mu_{M,high}(g_2), \mu_{A,low}(g_2), \mu_{A,medium}(g_2), \mu_{A,high}(g_2),$ $\mu_{C,low}(g_2), \mu_{C,medium}(g_2), \mu_{C,high}(g_2), \mu_{AV,low}(g_2), \mu_{AV,medium}(g_2), \mu_{AV,high}(g_2))$
...	...

2. If a candidate node t with a fuzzy set of data D verifies:
   (a) If the relative frequency of some class $C_k \in Average, Minimum$ in the dataset is over some threshold θ_r:

$$\frac{|D^{C_k}|}{|D|} \geq \theta_r \tag{13}$$

   (b) Or the cardinality of the dataset is under a given threshold:

$$|D| \leq \theta_n \tag{14}$$

   (c) Or there are no attributes for more classification
   Then it is a leaf node and the weight of each class in this leaf is assigned as the cardinality $|D^{C_k}|$ of the corresponding class C_k in such node.
3. Otherwise, the new decision node is constructed as follows, by selecting the attribute that maximizes the information gain $G(A_i, D)$. Therefore, for each attribute $A_i \in \{M, A, C, AV\}$ not considered before, calculate the information gain $G(A_i, D)$ (Eqs (15)-(19)) and select the attribute A_{max} that maximizes it:

$$G(A_i, D) = I(D) - E(A_i, D) \tag{15}$$

   where,

$$I(D) = -\sum_{k=1}^{n}(p_k * log_2 p_k) \tag{16}$$

$$E(A_i, D) = \sum_{j=1}^{m}(p_{ij} * I(D_{A_i,j})) \tag{17}$$

$$p_k = \frac{|D^{C_k}|}{|D|} \tag{18}$$

$$p_{ij} = \frac{|D_{A_i,j}|}{\sum_{l=1}^{m}|D_{A_i,l}|} \tag{19}$$

   Here $I(D)$ at Eq. (16) is the total entropy of certain dataset D, while $E(A_i, D)$ at Eq. (17) is the fuzzy classification entropy of the attribute A_i. p_k is the relative frequency of the class C_k in the dataset, and p_{ij} is the relative frequency of all objects within the branch associated to the corresponding linguistic label j and attribute A_i, into each class. $D_{A_i,j}$ is the fuzzy subset which membership is represented by the linguistic term $j \in \{low, medium, high\}$ linked to the group attribute $A_i \in \{M, A, C, AV\}$.

4. Once the attribute A_{max} that maximizes the information gain is chosen, **the decision node D is divided into three fuzzy subsets $D_{A_{max,low}}$, $D_{A_{max,medium}}$, $D_{A_{max,high}}$ according to such attribute**, each subset for each linguistic label that characterizes such attribute. The membership value of each group g to $D_{A_{max},j}$ ($j \in \{low, medium, high\}$), is then the product of the membership value of g to D, and the value $\mu_{A_{max,j}}(g)$ associated to A_{max} in D.
5. Generate new nodes t_1, t_2, t_3 for fuzzy subsets $D_{A_{max,low}}$, $D_{A_{max,medium}}$, $D_{A_{max,high}}$, labeling with each corresponding linguistic term $j \in \{low, medium, high\}$, to each edge that connect them with D.
6. For each fuzzy subset $D_{A_{max,low}}$, $D_{A_{max,medium}}$, $D_{A_{max,high}}$, repeat recursively this algorithm from step 2.

Once the fuzzy decision tree is generated, it will be used in the next phase of the proposal as the base of the fuzzy rules building and classification processes.

3.3. Fuzzy rules building and classification

Once the fuzzy decision tree is built, their branches lead to the creation of decision rules which are directly used for classification. The rules are formulated as follows, considering for every branch all the associated attributes and linguistic terms at the nodes from the root, to the leaf of the branch.

$$Rule\ R = \ If\ A_{i1}\ is\ j1\ and\ ...\ and\ If\ A_{in}\ is\ jn\ then\ Class = C_k\ with\ weight\ W_k \tag{20}$$

Here $A_{i1} \in \{M, A, C, AV\}$ is an attribute used for identifying groups, and $j1 \in \{low, medium, high\}$ is a linguistic term for representing the corresponding fuzzy set for characterizing such attribute, linked to the corresponding path in the inferred decision tree. $C_k \in \{Average, Minimum\}$ is the class label in the leaf node. W_k is the weight of the leaf node, calculate through the addition of the membership of all objects of class k at this node.

In this context, and assuming a new group g, the fuzzy classification is performed as follows:

1. Matching degree: Here, the activation degree of the if part for a rule R, with the group g, is calculated as:

$$\mu_R(g) = T(\mu_{A_{i1,j1}}(g), \mu_{A_{i2,j2}}(g), ..., \mu_{A_{in,jn}}(g)) \tag{21}$$

   where $\mu_{A_{i,j}}(g)$ is the membership degree of the value of the $A_i \in \{M, A, C, AV\}$ attribute of the example g with the fuzzy set associated to the same attribute A_i and the linguistic term $j \in \{low, medium, high\}$, at the corresponding antecedent of the rule R. T is a T-norm [38].
2. Association degree: The degree of the association of the group g with each rule R at the rule base and for the class k is computed as:

$$b_{Rk}(g) = T(\mu_R(g), W_k) \tag{22}$$

   where W_k is the weight of the rule R for the class k (i.e. the rule weight, see Eq. (20)). T is a T-norm [38].
3. Confidence degree: Finally, the confidence degree of each class is calculated by aggregating the association degrees of the rules of that class through the use of an operator T^*, being a T-conorm [38]:.

$$conf_k(g) = T^*(b_{1k}(g), b_{2k}(g), b_{3k}(g), ..., b_{Rk}(g)) \tag{23}$$

   Here $b_{Rk}(g)$, is the association degree of the group g to the class k according to the rule R.
   The group g is then classified as the class k with the highest confidence degree $conf_k(g)$, considering all the rules identified at the decision tree induction process.

Section 4 will demonstrate the use of the procedure presented here, in a GRS scenario.

3.4. Algorithmic overview of the approach

As summary, Algorithm 1 presents an overview of the current approach, receiving as input the set of groups G at the GRS, and the current group c for finding the most appropriate aggregation function, which is the output of the approach.

At first, Lines 4-12 are focused on the dataset building phase, calculating for each group the values of the four attributes for characterizing them, as well as the aggregation function that performs better in a recommendation generation process. Finally, a tuple with this information is added to a dataset D (Line 12) that is used in the subsequent stages of the method.

Furthermore, Line 13 obtains the fuzzy decision tree T using such dataset D. Subsequently, such tree is used for building the classification rules that will allow to obtain, for any group of users characterized by the four mentioned attributes, the most appropriate aggregation function to be used (Lines 14-16).

Finally, such set of rules is used for finding the referred aggregation function for the current group c (Line 17), retrieving it as the output of the method (Line 18).

Algorithm 1 has presented all the phases of the proposal in order to expose a compact overview of its working principle. However, it is also worthy to note that in practice, the dataset building, tree induction, and rules building phases (Lines 4-16) can be executed previously in an offline stage, and stored the rules set R. In this way, in the real-time recommendation generation for a specific group c, it would be directly executed the fuzzy classification step (Line 17), using the stored rules set.

Algorithm 1 Algorithmic overview of the approach.

1: **procedure** FUZZY CB-GRS(a, b)
2: Input: c-currentGroup, G-set of groups
3: Output: selectedAggFunction- Aggregation function to be used in the current group
4: **for all** group g in G **do**
5: M_g = MinimumCorrelation(g)
6: A_g = AmountGroupR(g)
7: C_g = CoRated(g)
8: AV_g = RatingAverage(g)
9: Calculate recommendation performance with average preference aggregation for group g
10: Calculate recommendation performance with minimum preference aggregation for group g
11: Assign $Class_g$ as the aggregation scheme that performs better
12: Add tuple (M_g, A_g, C_g, AV_g, $Class_g$) to the dataset D
13: T=ObtainFuzzyDecisionTree(D)
14: **for all** path p from root to the leaves nodes in T **do**
15: Build the associated classification rules r
16: Add r to the rules set R
17: selectedAggFunction= Rule-basedClassifier(R, c)
18: return selectedAggFunction

In the next sections it will be presented a case study and the evaluation of an experimental protocol associated to the current proposal.

4. Case study

This subsection develops and describes a case study showing how the algorithm presented in the previous subsection can be used for the dynamic selection of the most appropriate function for individual recommendation aggregation.

4.1. Fuzzy decision tree induction

At first, Table 4 presents a dataset that can be obtained through the methodology at Section 3.1, that contains five groups. In this case, for simplicity it is composed of 3 attributes which are minimum correlation (MinCorr, i.e. M), amount of ratings in the group (AmountGroupR, i.e. A) and average amount of co-rated items across users (Co-RatedAvg, i.e. C). Two classes are considered, the average aggregation and the minimum aggregation (Avg, and Min). In the next future it will be additionally considered more sophisticated aggregation schemes such as the Additive Utilitarian hybridized with the Approval Voting [49], as well as the Agreement without Uncertainty approach [49]. However, they are out of the scope of this work.

Table 4 shows the value of each attribute at the five objects. Furthermore, it contains the membership value of each object, to the three previously mentioned fuzzy sets A_{low}, A_{medium}, A_{high}, for each attribute A (referred at the previous Section 3.2). Subsequently, in this section is developed the decision tree induction associated to such dataset, which is presented in Fig. 9. Furthermore, as parameters it is considered $\theta_r = 0.9$, (i.e. a node is not expanded when the cardinality of some class over the total cardinality exceeds 0.9); and $\theta_n = 0.01$, (i.e. a node is not expanded when its fuzzy cardinality in under 0.01). We use this value for θ_n for guaranteeing the generation of a decision tree as large as possible, with demonstrative proposals. Larger values of θ_n would lead to a less-expanded decision tree. However, they are not included here due to space reasons.

Root node: Following the steps of the decision tree induction procedure presented above, at first it is considered the current root node with the five objects with membership $\mu_n(g) = 1$. Therefore, the calculation of information amount $I(D)$ of such node (Eq. (16)) and the information gain $G(A, D)$ at each attribute A (Eq. (15)), lead to the following results:

$$I(D) = 0.971, \quad G(M, D) = 0.2143, \quad G(A, D) = 0.2897, \quad G(C, D) = 0.1301 \tag{24}$$

Here the attribute that maximizes the information gain is the amount of ratings in the group (AmountGroupR, A). Then, according to the steps 4 and 5 at Section 3.2, the set of objects in the current root node is divided in three fuzzy subsets, characterized by the membership functions associated to such attribute A. Three new nodes are respectively created for such subsets. At Fig. 9, these nodes are labeled as (a_{low}), (a_{med}), and (a_{high}).

Node a_{low}: Analyzing the node a_{low}, characterized by the objects g_2 ($\mu_n(g_2) = 0.82$) and g_4 ($\mu_n(g_4) = 1$). Here it is analyzed the information amount $I(D_{A_{low}})$ of this node, as well as the information gain for the remaining attributes M and C:

$$I(D_{A_{low}}) = 0.993, \quad G(M, D_{A_{low}}) = 0.993, \quad G(C, D_{A_{low}}) = 0.0998 \tag{25}$$

Table 4
Case study for the dynamic selection of the aggregation function.

	MinCorr(M)	AmountGroupR(A)	Co-RatedAvg(C)	Class
g_1	0.3 $\mu_{M,low}(g_1) = 0.24$ $\mu_{M,medium}(g_1) = 0.76$ $\mu_{M,high}(g_1) = 0$	300 $\mu_{A,low}(g_1) = 0$ $\mu_{A,medium}(g_1) = 0.9$ $\mu_{A,high}(g_1) = 0.1$	20 $\mu_{C,low}(g_1) = 0$ $\mu_{C,medium}(g_1) = 0.9$ $\mu_{C,high}(g_1) = 0.1$	Avg
g_2	0.7 $\mu_{M,low}(g_2) = 0$ $\mu_{M,medium}(g_2) = 0$ $\mu_{M,high}(g_2) = 1$	150 $\mu_{A,low}(g_2) = 0.82$ $\mu_{A,medium}(g_2) = 0.18$ $\mu_{A,high}(g_2) = 0$	10 $\mu_{C,low}(g_2) = 0.82$ $\mu_{C,medium}(g_2) = 0.18$ $\mu_{C,high}(g_2) = 0$	Avg
o_3	0.05 $\mu_{M,low}(g_3) = 1$ $\mu_{M,medium}(g_3) = 0$ $\mu_{M,high}(g_3) = 0$	450 $\mu_{A,low}(g_3) = 0$ $\mu_{A,medium}(g_3) = 0$ $\mu_{A,high}(g_3) = 1$	30 $\mu_{C,low}(g_3) = 0$ $\mu_{C,medium}(g_3) = 0$ $\mu_{C,high}(g_3) = 1$	Min
g_4	0.1 $\mu_{M,low}(g_4) = 0.84$ $\mu_{M,medium}(g_4) = 0.16$ $\mu_{M,high}(g_4) = 0$	120 $\mu_{A,low}(g_4) = 1$ $\mu_{A,medium}(g_4) = 0$ $\mu_{A,high}(g_4) = 0$	8 $\mu_{C,low}(g_4) = 1$ $\mu_{C,medium}(g_4) = 0$ $\mu_{C,high}(g_4) = 0$	Min
g_5	0.5 $\mu_{M,low}(g_5) = 0$ $\mu_{M,medium}(g_5) = 0.62$ $\mu_{M,high}(g_5) = 0.38$	400 $\mu_{A,low}(g_5) = 0$ $\mu_{A,medium}(g_5) = 0.3$ $\mu_{A,high}(g_5) = 0.7$	15 $\mu_{C,low}(g_5) = 0.36$ $\mu_{C,medium}(g_5) = 0.64$ $\mu_{C,high}(g_5) = 0$	Min

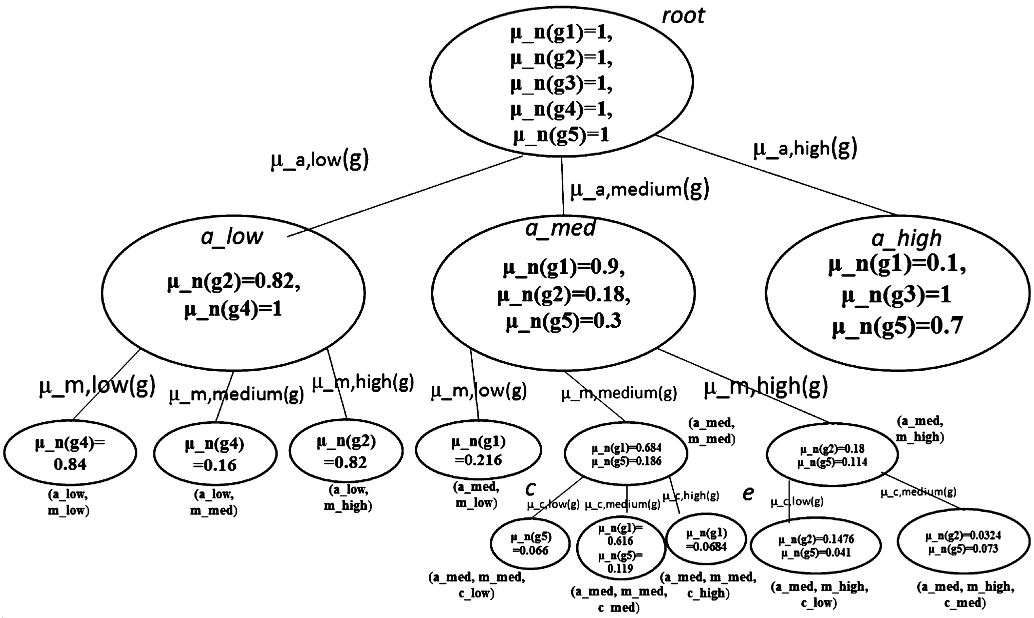

Fig. 9. Fuzzy decision tree of the dataset at Table 4, for the dynamic selection of the aggregation function.

which lead to the selection of the attribute M at this stage.

This leads to the expansion of the nodes (a_{low}, m_{low}), (a_{low}, m_{med}), and (a_{low}, m_{high}) (see Fig. 9).

These three nodes have only one associated object, therefore the process stops at this stage considering that here the cardinality of its associated class is 1. Such nodes are then considered as leaves nodes, having as their corresponding classes, the class associated to their objects.

Node a_{med}**:** This node is characterized by the objects $(\mu_n(g_1) = 0.9)$, g_2 $(\mu_n(g_2) = 0.18)$, and g_5 $(\mu_n(g_5) = 0.3)$. In this node, the attributes M and C are also analyzed:

$$I(D_{A_{med}}) = 0.755, \quad G(M, D_{A_{med}}) = 0.0779, \quad G(C, D_{A_{med}}) = 0.0542 \tag{26}$$

which lead, in a similar way to node a_{low}, to the selection of the attribute M at this stage.

This leads to the expansion of the nodes (a_{med}, m_{low}), (a_{med}, m_{med}), and (a_{med}, m_{high}) (see Fig. 9).

Table 5
Rules associated to the decision tree at Fig. 9.

R_1	if A is low and M is low, then Class=Min with W=0.84
R_2	if A is low and M is medium, then Class=Min with W=0.16
R_3	if A is low and M is high, then Class=Avg with W=0.82
R_4	if A is medium and M is low, then Class=Avg with W=0.216
R_5	if A is medium and M is medium and C is low, then Class=Min with W=0.066
R_6	if A is medium and M is medium and C is medium, then Class=Avg with W=0.616, and Class=Min with W=0.119
R_7	if A is medium and M is medium and C is high, then Class=Avg with W=0.0684
R_8	if A is medium and M is high and C is low, then Class=Avg with W=0.1476, and Class=Min with W=0.041
R_9	if A is medium and M is high and C is medium, then Class=Avg with W=0.0324, and Class=Min with W=0.073
R_{10}	if A is high, then Class=Avg with W=0.1, and Class=Min with W=1.7

Table 6
Classification example.

	MinCorr(M)	AmountGroupR(A)	Co-RatedAvg(C)
o	0.4 $\mu_{M,low}(o) = 0$ $\mu_{M,medium}(o) = 0.92$ $\mu_{M,high}(o) = 0.08$	300 $\mu_{A,low}(o) = 0$ $\mu_{A,medium}(o) = 0.9$ $\mu_{A,high}(o) = 0.1$	25 $\mu_{C,low}(o) = 0$ $\mu_{C,medium}(o) = 0.46$ $\mu_{C,high}(o) = 0.54$

Here the node (a_{med}, m_{low}) has only one associated object, therefore the process stops at this stage. On the other hand, the remaining two nodes are analyzed as follows.

Node (a_{med}, m_{med}): This node is characterized by two objects (g_1 ($\mu_n(g_1) = 0.684$), g_5 ($\mu_n(g_5) = 0.186$)). It is then expanded through the remaining attribute C (see Fig. 9), composing the three nodes $(a_{med}, m_{med}, c_{low})$, $(a_{med}, m_{med}, c_{med})$, and $(a_{med}, m_{med}, c_{high})$. These new nodes are not expanded because there are not any more attributes.

Node (a_{med}, m_{high}): This node is characterized by two objects (g_2 ($\mu_n(g_2) = 0.18$), g_5 ($\mu_n(g_5) = 0.114$)). It is then expanded through the remaining attribute C (see Fig. 9), composing the two nodes $(a_{med}, m_{high}, c_{low})$ and $(a_{med}, m_{high}, c_{med})$. These new nodes are not expanded because there are not any more attributes. The node $(a_{med}, m_{high}, c_{high})$ is not included because it does not contain any object with membership values different from zero (see Fig. 9).

Node a_{high}: This node is characterized by the objects g_1 ($\mu_n(g_1) = 0.1$), g_3 ($\mu_n(g_3) = 1$), and g_5 ($\mu_n(g_5) = 0.7$). At such node, it is verified the stop condition (a) at the second step of the procedure, regarding that for the class avg, $|D^{avg}|/|D| = 0.944 > 0.9$, assuming that $\theta_r = 0.9$. Therefore this node is not expanded.

This stage finishes the induction of the fuzzy decision tree presented at Fig. 9. According to Section 3.3, such tree leads to the rules based on the class of the objects at the leaves nodes, presented at Table 5

4.2. Fuzzy classification

Based on the rules presented at Table 5, as example it will be classified the group o presented at Table 6. The first row of the table presents the values of the attributes of the object, while the remaining rows present the membership values associated to the three fuzzy sets (Fig. 8) that characterize each attribute according to the current dataset presented at Table 6.

Subsequently, Table 7 illustrates the three stages of the classification procedure at Section 3.3. At the first stage, it is calculated the matching degree of the object with the antecedents of the rules. At Table 7, it is only presented the rules with a matching degree different from 0, which are R_6, R_7, R_9, and R_{10}. Furthermore, in the next stage it is calculated the association of the object with the corresponding class linked to the rule. Here it is worthy to note that in three of the four considered rules, there is a contribution value of the rule to both classes, therefore both values are individually computed at each individual class processing. Both stages depend on a T-norm, being used the *min* operator in this scenario.

Table 7
Classification procedure.

Stage 1	$\mu_{R_6}(o) = 0.46$, $\mu_{R_7}(o) = 0.54$, $\mu_{R_9}(o) = 0.08$, $\mu_{R_{10}}(o) = 0.1$
Stage 2	**Class Avg** $b_{R_6,avg} = T(\mu_{R_6}(o), RW_6^{avg}) = T(0.46, 0.616) = 0.46$ $b_{R_7,avg} = T(\mu_{R_7}(o), RW_7^{avg}) = T(0.54, 0.0684) = 0.0684$ $b_{R_9,avg} = T(\mu_{R_9}(o), RW_9^{avg}) = T(0.08, 0.0324) = 0.0324$ $b_{R_{10},avg} = T(\mu_{R_{10}}(o), RW_{10}^{avg}) = T(0.1, 0.1) = 0.1$ **Class Min** $b_{R_6,min} = T(\mu_{R_6}(o), RW_6^{min}) = T(0.46, 0.119) = 0.119$ $b_{R_9,min} = T(\mu_{R_9}(o), RW_9^{min}) = T(0.08, 0.073) = 0.073$ $b_{R_{10},min} = T(\mu_{R_{10}}(o), RW_{10}^{min}) = T(0.1, 1.7) = 0.1$
Stage 3	$conf_{avg}(o) = T^*(b_{R_6,avg}, b_{R_7,avg}, b_{R_9,avg}, b_{R_{10},avg}) = 0.46$ $conf_{min}(o) = T^*(b_{R_6,min}, b_{R_9,min}, b_{R_{10},min}) = 0.119$ $class_o = avg$

Finally, in the last stage at Table 7, it is calculated the confidence degree of each class, by using a T-conorm for joining the association degree of each rule, associated to each independent class. Here it is used the *max* operator as T-conorm. As final result it is reached that the avg class has a higher confidence; therefore the object o is classified with the avg class.

Then, it means that the avg aggregation approach seems to be the best option for aggregating individual recommendations in the group characterized by the attributes associated to the object o.

5. Experiments

This section is focused on evaluating the content-based GRS presented across this paper. At first, they are presented the datasets (Section 5.1), the evaluation metric (Section 5.2), and the evaluation protocol used in the experimentation (Section 5.3). Subsequently, for each dataset it is presented an exploratory study on the values of such attributes across the datasets (Section 5.4); and it is then evaluated the performance of the proposal comparing it against baselines (Section 5.5). A discussion on the results is also included (Section 5.6), as well as future works to expand the current proposal (Section 5.7).

5.1. Datasets

This work is supported by two recognized datasets that have been used for studying content-based recommendation systems [41].

- **Movielens 100K,** having 943 users, 1682 movies, and 100000 ratings in the range $[1, 5]$ [26]. Movies are classified considering 19 possible genres, including Action, Sci-Fi, Comedy, Adventure, etc. In this context, the item profile is represented as a binary vector composed of 19 dimensions, having 1 whether the corresponding genre is associated to the movie, and 0 otherwise. This dataset is used to evaluate the binary item profile approach.
- **HetRec,** which is also a well known dataset in RS research that considers heterogeneous item profiling [7]. From this dataset the qualitative attributes genre, director, country; the quantitative audience score; and the year, are used for representing a multivalued item profile. In the case of the year, operations like average or mode have no sense, and therefore we represent it as a qualitative value. Furthermore, in this case we execute our evaluation with the first 300 users in the dataset.

In both cases we use the same data employed by Pérez-Almaguer et al. [41], which is the former work used a base for the development of the current proposal, and that will be compared with our work in the next section. Such use of the same data, guarantees a fair comparison of both approaches.

5.2. Evaluation metric

In order to evaluate the proposals, it will be considered the precision metric, which has been used previously for evaluating the top n item recommendation task in group recommender systems [5,31]. Here it is important to point out that this metric is used in two different scenarios: 1) in the dataset building stage of the decision tree induction process for identifying which aggregation scheme performs the best for each group, and 2) in the core CB-GRS approach that already integrates the dynamic selection process, to evaluate the recommendation accuracy of the current group. In the next future, other relevant metrics such as NDCG will be also used; however at this stage we decided to be focused in one specific metric considering its dual use across the framework. A parallel use of a second metric here, goes beyond our current objective, which is the presentation of a proof-of-concept of the dynamic aggregation approach in GRS. In addition, it would lead to further issues such as determining which metrics use in the first and the second mentioned scenarios, and how to evaluate their effects in the recommendation performance.

For each list of top n recommended items, Precision [25] is defined as the ratio between the amount of recommended items that were actually preferred by the current user, and the overall amount of recommended items (in this case k).

$$Precision = \frac{|recommended\ items \cap preferred\ items|}{|recommended\ items|} \tag{27}$$

To complete the Precision task, all the items in the test set of each user are taken into account for recommendation generation, in a similar way to previous works focused on the same recommendation task [5,19,36]. In addition, for Precision calculation it is used a preference threshold $pref$ that considers preferred items as those that verify $r_{ui} \geq 4$, which is a common criterion for this threshold selection [43].

5.3. Experimental protocol

We evaluate the proposal across the following steps [13,25]:

- Initially, the set of ratings associated to each user profile is randomly split into the user-associated training and test sets. The final training and test sets, are then built by merging the training and test set of each user.
- We build user groups of different sizes, following a group formation criteria that will be explained below.
- For the whole training dataset composed of all the groups, we execute the procedure for getting the classification rules, supported by the fuzzy decision tree induction (Fig. 6). In the current context we match the group sampling required for the tree induction, with the whole group population.
- For each group, we apply the proposed framework over the training data of their users including the dynamic selection step supported by the obtained classification rules. As result, the top n recommended items for the group are reached.
- The accuracy of the top n recommendation list is evaluated through the accuracy metric, by contrasting it with the items currently associated to each user in the test set, considering as ground truth the preferences associated to such test set. The average accuracy values for all groups are finally computed.

Furthermore, several approaches have been considered in the literature for group composition. Here we use a criterion that guarantees that groups' members always keep some characteristics in common [5,31], and specifically we follow the criteria referred at Kaššák et al. [31], that consider groups composed by individuals that have commonly rated the same set of items. Here, we regard item sets of size 5 located in the ratings test set, considering that in a practical experimental scenario is difficult the composition of groups with a larger size, and that a size below 5 does not represent a compact group able to be properly characterized through the attributes proposed at the current paper (see Section 3.1).

We consider groups with sizes 3 and 4 members depending of each experimental scenario. For each case, 20 groups that guarantee the fulfillment of this criteria [31] are built. For each group, the top n recommendation list is generated by varying n in the range $[1, 5]$ with step 1, and also in the range $[5, 20]$ with step 5. This process is repeated 10 times, averaging the results.

5.4. Exploratory study of the groups' attribute selection

Section 3.1 detailed the four attributes that will be used for group characterization. Such attributes are 1) the minimum correlation between any pair of group members, 2) the amount of ratings across the group, 3) the amount of co-rated items by all the users, and 4) the rating average of the group. This section will present an exploratory study of the groups' attribute selection at the used datasets.

5.4.1. Movielens

Initially, Figs. 11–13 present the frequency of the values of the four considered attributes for all the groups in the dataset Movielens. As mentioned, such attributes are: 1) the minimum correlation between any pair of group members, 2) the amount of ratings across the group, 3) the amount of co-rated items by all the users, and 4) the rating average of the group.

Minimum correlation between group members: In the case of the attribute *minimum correlation between group members* (Fig. 10), it can be observed that the sampled groups present values distributed across all the possible correlation values, even though a larger number of groups have correlation values in the range $[-0.5, 0]$. In addition, there were several groups with correlation values equal to zero and one. The tendency to have a balanced distribution across all the possible correlation values, guarantees in advance a better attribute exploitation in the decision tree induction process (see Section 3.2). In this way it guarantees that the available data would cover most of the possible membership values associated to the fuzzy set low, $medium$, and $high$, linked to the current attribute.

Amount of group ratings: In the case of the attribute *amount of group ratings* (Fig. 11), it can be observed that most of groups have an attribute value in the range $[80, 400]$, being concentrated in the range $[80, 160]$. Furthermore, it can be

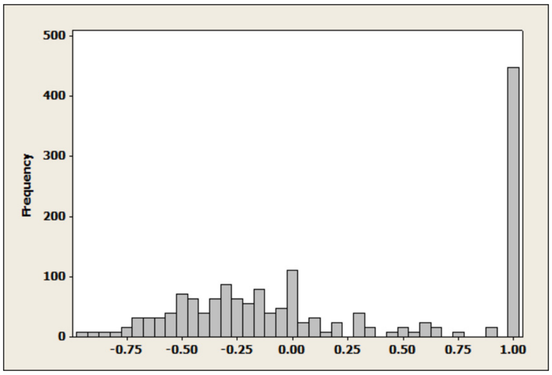

Fig. 10. Histogram presenting the frequency of the values for the attribute *minimum correlation between group members*, across all the sampled groups in Movielens 100K.

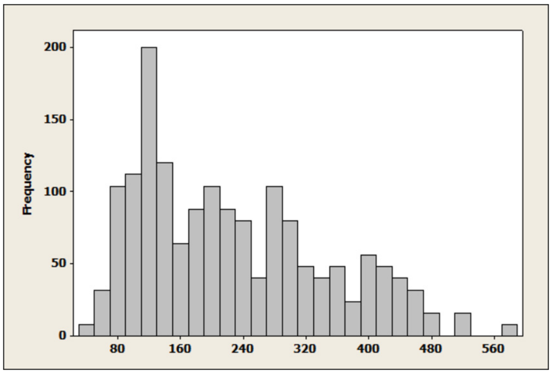

Fig. 11. Histogram presenting the frequency of the values for the attribute *amount of group ratings*, across all the sampled groups in Movielens 100K.

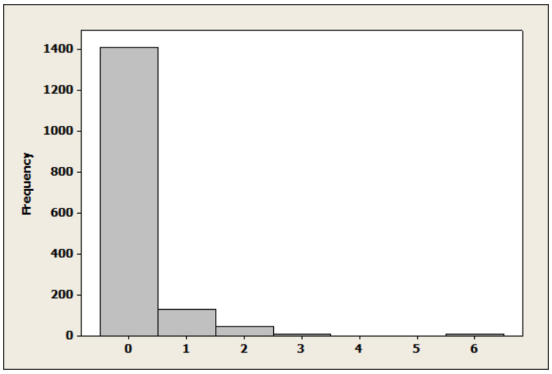

Fig. 12. Histogram presenting the frequency of the values for the attribute *amount of co-rated items*, across all the sampled groups in Movielens 100K.

identified that there are several groups with a high membership value associated to the low and $medium$ fuzzy sets related to such attribute (see Section 3.2).

Amount of co-rated items: Subsequently, Fig. 12 shows the behavior of the attribute *amount of co-rated items* across all groups. As could be initially expected, there is a strong tendency in the groups to have low values of this attribute. Specifically, most of the explored groups have an amount of co-rated items under four. This imbalance in the frequency behavior suggests the necessity of exploring further membership functions in the future, beyond the function presented at Section 3.2, to be used in scenarios like this one. Furthermore, in the next Section 5.4.2, it will be showed how the frequency values of this attribute for composed groups at the HetRec dataset are different in relation to the presented here for Movielens.

Rating average of the group: Finally, Fig. 13 shows the behavior of the attribute *rating average of the group* across all groups. Here, most of groups lie in the range $[3, 3.8]$, presenting also some imbalance even though it is not as large as in the attribute *amount of co-rated ratings*. Despite of such imbalance, here the generic fuzzy sets presented in Section 3.2

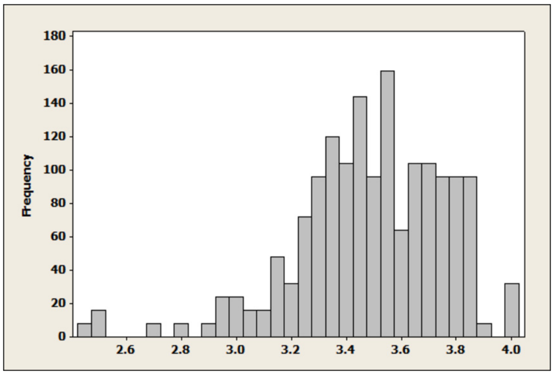

Fig. 13. Histogram presenting the frequency of the values for the attribute *rating average of the group*, across all the sampled groups in Movielens 100K.

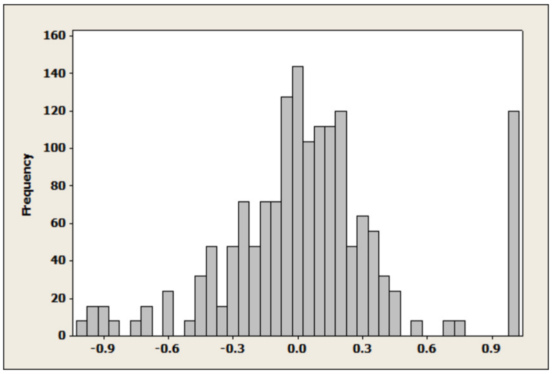

Fig. 14. Histogram presenting the frequency of the values for the attribute *minimum correlation between group members*, across all the sampled groups in HetRec.

for characterizing these values, are able to discriminate among groups with high membership values for the sets *medium* and *high*. We remark that the use of more sophisticated membership functions as in Yera et al. [51], tied to some specific attributes distribution values, are out of the scope of this paper.

5.4.2. HetRec

This subsection presents the frequency of the values of the four considered attributes (Section 3.1), for all the groups in the dataset HetRec using the same scheme previously used for the other dataset (Section 5.4.1). Initially, Figs. 15–17 present the frequency of the values of the four attributes for all the groups.

Minimum correlation between group members: In the case of the attribute *minimum correlation between group members*, Fig. 14 presents a behavior that is similar to the associated to the Movielens 100K dataset at Fig. 10, in the sense that the groups contained values distributed across all the possible correlation values, also with several groups with correlation values equal to zero and one. However, in contrast to the former dataset, in this case there is an important amount of positive correlation values; lying in the range $[0, 0.3]$.

Amount of group ratings: In the case of the attribute *amount of group ratings* (Fig. 15), it can be observed that most of groups have an attribute value in the range $[20, 800]$, being common the frequency of groups in the range $[300, 400]$. Therefore, such groups will have a higher membership value to the fuzzy set low, and a lower membership value to the fuzzy set *medium*, associated to this attribute (see Section 3.2). These results are different to the behavior of the previously analyzed Movielens dataset, where the amount of ratings of the built groups, were more distributed across the minimum and maximum possible value for the attribute.

Amount of co-rated items: Subsequently, Fig. 16 presents the frequency values associated to the attribute *amount of co-rated items*. It is worthy to note that in contrast to the Movielens dataset (Fig. 12), the amount of co-rated items for groups was not concentrated in the smallest possible values, and here it is reported several groups with values in the range $[1, 18]$. However, it is also reported an important imbalance across values, and for this reason other membership functions for the currently defined fuzzy sets (see Section 3.2) should be explored in the future for this kind of attributes, as was also pointed out in Movielens.

Rating average of the group: Finally, Fig. 13 shows the behavior of the attribute *rating average of the group* across all groups. In contrast to the Movielens dataset, here all the values are concentrated in the well-defined range $[2.85, 4]$. There-

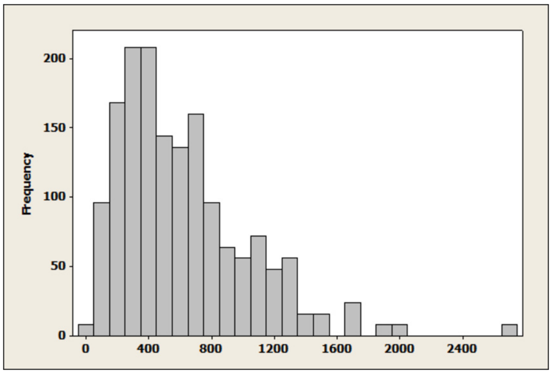

Fig. 15. Histogram presenting the frequency of the values for the attribute *amount of group ratings*, across all the sampled groups in HetRec.

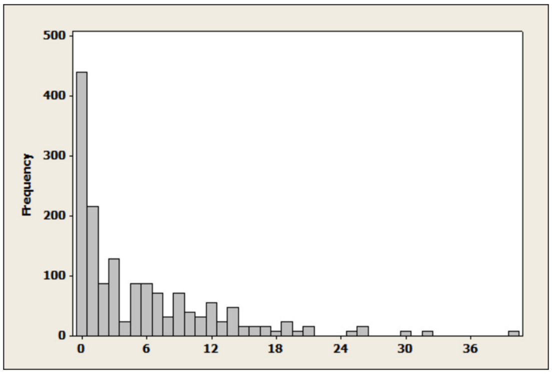

Fig. 16. Histogram presenting the frequency of the values for the attribute *amount of co-rated items*, across all the sampled groups in HetRec.

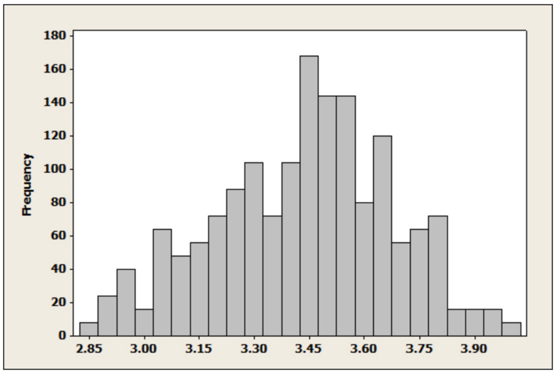

Fig. 17. Histogram presenting the frequency of the values for the attribute *rating average of the group*, across all the sampled groups in HetRec.

fore, the proposed fuzzy sets are able to represent better each group particularities across the group's membership values to the sets low, $medium$, and $high$. Here the higher frequencies were associated to a rating average around 3.45.

Overall, this exploratory study suggested that the identified attributes are able to discriminate among groups, and that such discrimination can be useful for selecting the most appropriate CB-GRS approach.

5.5. Results

This section presents the results of the evaluation of the proposal, in both Movielens 100K and MovieTweeting datasets. This evaluation is focused on: 1) identifying the impact of the new proposal that incorporates the fuzzy decision tree approach, by comparing its performance with previous baselines that do not consider the fuzzy decision tree, and 2) measuring the performance of the proposal when one of the four identified group attributes (see Section 3.1) is not taken into account for the decision tree building. This second criteria contribute to quantify the effect of each individual group attribute, in the full proposal.

Table 8
Evaluation of the proposal and comparison with baselines avg and min, presented at [41]. Movielens 100K dataset. Precision value.

top N	1	2	3	4	5	10	15	20
Approaches that does not consider the fuzzy decision tree induction process								
avg (baseline)	0.5787	**0.5844**	0.5788	0.5684	0.5740	0.5681	0.5628	0.563
min (baseline)	0.5813	0.5725	0.5829	0.5841	0.5845	0.5754	0.5708	**0.5713**
Approaches incorporating the fuzzy decision tree induction process								
dyn	**0.6025**	0.5806	**0.5879**	0.5844	0.5855	0.5760	0.5686	0.5685
dyn not AV	0.600	0.5788	0.5850	0.5838	0.5850	0.5763	0.5698	0.5693
dyn not C	0.5838	**0.5844**	**0.5879**	0.5841	0.5850	0.5744	0.5669	0.5651
dyn not M	0.5963	0.5763	0.5846	**0.5853**	**0.5873**	**0.5785**	**0.5712**	0.5712
dyn not A	0.5925	0.5725	0.5825	0.5813	0.5833	0.5769	0.5703	0.5699

Specifically, seven approaches will be evaluated:

- The current proposal, detailed at Section 3 (dyn).
- The current proposal, detailed at Section 3, but without considering the attribute *minimum correlation between any pair of group members* (dyn not M).
- The current proposal, detailed at Section 3, but without considering the attribute *amount of ratings of the group* (dyn not A).
- The current proposal, detailed at Section 3, but without considering the attribute *amount of co-rated items* (dyn not C).
- The current proposal, detailed at Section 3, but without considering the attribute *rating average* (dyn not AV).
- As baseline, the former CB-GRS based on recommendation aggregation and user-item matching values [41], always using average aggregation (avg) and without the use of the fuzzy decision tree.
- As baseline, the former CB-GRS based on recommendation aggregation and user-item matching values [41], always using minimum aggregation (min), and without the use of the fuzzy decision tree.

5.5.1. Movielens

This subsection presents the results associated to the dataset Movielens 100K.

Here the proposal is evaluated considering as parameters $\theta_r = 0.9$ and $\theta_n = 0.01$, i.e. the stop conditions are executed once the relative frequency of some class in the current node is equal or higher that 0.9, or when the cardinality of the set as such node is under 0.01, see Section 3.2. Further executions were performed for other values of θ_r and θ_n, reporting here the values that lead to a better performance.

The results are obtained for groups of size 4; and the size of the top n recommendation lists were in the range $[1, 5]$ with step 1, and $[5, 20]$ with step 5.

Table 8 shows these results, being differentiated those associated with our proposal that incorporates the fuzzy decision tree induction, and those associated to the baseline that does not consider it. Here it is presented that for 6 of the 8 experimental scenarios the proposal is able to outperform the baselines, and for n=2 and n=20, it reaches a similar behavior to the average and minimum approach respectively. Furthermore, it is relevant to mention that while for n=1 and n=3 the best performance is reached when the four group attributes are used, for the other scenarios it is reached with the exclusion of some attribute. Specifically, for n=2 the best result was obtained with the exclusion of the attribute *amount of co-rated items*. In addition, for n= $\{4, 5, 10, 15\}$, the best results were obtained with the exclusion of the attribute *minimum correlation between any pair of members*, even though the values of this attribute in the groups, have a tendency to a fair distribution across all the possible attribute values (Fig. 10).

Overall, the results evidence that for the Movielens dataset, the proposal is able to identify the best aggregation function to use in a CB-GRS based on recommendation aggregation, considering that it notably outperforms two baselines that always use the average and minimum aggregation.

5.5.2. HetRec

This subsection presents the results associated to the dataset HetRec.

The proposal is evaluated considering as parameters $\theta_r = 0.9$ and $\theta_n = 0.01$, and compared with the baselines also considered for the Movielens dataset. In a similar way to Movielens, further executions were performed for other values of θ_r and θ_n, reporting here the values that lead to a better performance.

The results are obtained for groups of size 3 considering that larger groups were not able to obtain due to the sparsity of the dataset. Furthermore, the sizes of the top n recommendations lists were in the range $[1, 5]$ with step 1, and $[5, 20]$ with step 5.

Table 9 presents the evaluation results, being differentiated those associated with our proposal, and those associated to the baseline that does not consider the fuzzy decision tree induction. In this case, for all the experimental scenarios, the new proposal outperforms the baselines. Specifically, for $n = \{3, 4, 5\}$ the use of the four group attributes for building

Table 9
Evaluation of the proposal and comparison with baselines avg and min, presented at [41]. HetRec dataset. Precision value.

top N	1	2	3	4	5	10	15	20
Approaches that does not consider the fuzzy decision tree induction process								
avg (baseline)	0.5050	0.5075	0.5039	0.5000	0.5013	0.4957	0.4952	0.4943
min (baseline)	0.5700	0.5483	0.5417	0.5358	0.5297	0.5080	0.5024	0.4994
Approaches incorporating the fuzzy decision tree induction process								
dyn	0.5817	0.5483	**0.5422**	**0.5363**	**0.5299**	0.5083	0.5029	0.4989
dyn not AV	**0.5833**	**0.5492**	0.5406	0.5350	0.5277	0.5080	**0.5031**	0.5000
dyn not C	0.5817	0.5467	0.5367	0.5333	0.5263	**0.5090**	0.5024	**0.5003**
dyn not M	0.5700	0.5500	**0.5422**	0.5354	0.5280	0.5083	**0.5031**	0.4993
dyn not A	0.5783	0.5483	0.5417	0.5358	0.5297	0.5080	0.5024	0.4994

the fuzzy decision tree leads to the best performance. Furthermore, for $n = \{1, 2, 15\}$, such performance was achieved by discarding the attribute *rating average*, and for two cases ($n = \{10, 20\}$) it was achieved by discarding the *amount of co-rated items*.

Overall, in this dataset is more clear the superiority of the proposal over the baselines, in relation to the previous dataset Movielens. Furthermore, it is also worthy to remark that there was not a specific group feature which exclusion globally leads to a performance improvement, across the different sizes of the recommendation lists.

5.6. Final discussion

This paper has introduced a content-based GRS framework, based on recommendation aggregation, and focused on performing a dynamic selection of the most appropriate aggregation functions according to the nature of the active group. The development of the proposal as well as the experimentation, leads to the following findings:

- The analysis of the literature related to content-based GRS, suggested that as far as we know, there is not a direct antecedent focused on the use of machine learning techniques, to learn knowledge from the groups' behavior and therefore using this knowledge for improving the recommendation generation for such groups.
- The exploratory analysis of the four group attributes considered in this work, which are the minimum correlation between any pair of group members, the amount of ratings across the group, the amount of co-rated items by all the users, and the rating average of the group; shows that they are able to characterize groups' behavior, even though it was with a larger or lesser success depending on the nature of the data.
- As could be expected, an exploratory analysis using frequency histograms, shows that there are different imbalance levels across the groups attributes. In cases such as the minimum correlation between group members in both datasets, it can be observed some balance across the groups regarding the frequency values of this attribute. In contrast, in other cases such as the amount of co-rated items or the amount of groups' ratings in HetRec, there are many groups that share the same or similar attribute values. This fact would need the introduction in this framework, of more sophisticated membership functions beyond the proposed in Fig. 8, for a better characterization of the data in the fuzzy decision tree induction process.
- The experimental results evidence a positive performance for Movielens 100K and HetRec datasets. In both cases, the new proposal leads to an improvement of the recommendation performance, showing the suitability of our approach focused on using a fuzzy decision tree for dynamically selecting the most appropriate aggregation function in a CB-GRS scenario, based on the groups' features. Specifically, it was evidenced that a CB-GRS with such dynamic selection, outperforms two CB-GRSs that always use average and minimum respectively, as aggregation operators.
- From a general viewpoint, the results show that the incorporation of a fuzzy decision tree in a content-based group recommendation model, for supporting the recommendation generation process, is able to improve the recommendation generation performance. In this way, for the Movielens dataset it was able to reach a Precision value up to 0.6025, while the baselines that does not use the fuzzy decision tree reach up to 0.5845. In the HetRec dataset, the proposal reaches a Precision up to 0.5833, while the baselines reach up to 0.5700.
- Overall, the experimental results show that the proposal can serve as a starting point for developing a new research branch focused on the dynamic selection of the most appropriate components of a GRS framework, taking as input some attributes of the group.

5.7. Future works

The aim of this research paper is to be an starting point in the research branch related to the dynamic selection of the components of a GRS, as it has been previously commented. At this moment, the next future work to continue this research would be:

- The exploration of more sophisticate t-norms, t-conorms, and membership functions for the fuzzy sets low, $medium$, and $high$ (Fig. 8), that reflect better the nature of data. In the current work, we have used the well-recognized triangular membership functions as the basic approach. However, they could not be the most appropriate for some scenarios. As future work it will be explored the role of trapezoidal and sigmoidal membership functions for boosting or decreasing the effect of some group's attributes values, in the group's membership to the nodes in the induced decision tree. Furthermore, other t-norms (e.g. product) and t-conorms (e.g. probabilistic sum), will be also explored [54,20].
- The use of further schemes for calculating the relevance of the item profiles for each corresponding user, in the inner content-based recommendation approach. Being use in the current scenario the cosine measure as the reference approach (see Section 2.1), in the next works it will be considered more sophisticated schemes incorporating further knowledge sources (e.g. ontologies, linked open data cloud, other graph-based structures, etc) [24].
- The use of feature engineering approaches for a better characterization of the identified features, as well as the extracting of other features. Feature management comprises a wide range of techniques that could be applied here, such as feature weighting, or the discovery of latent features [32].
- The use of other supervised classifiers beyond the fuzzy decision tree-based. Fuzzy decision trees have been currently used as a white-box and effective classifier. However, it is interesting to explore here other well-recognized alternatives, such as multi-classifiers, support vector machines, or deep learning-based classifiers [3].
- The use of the presented framework in other GRS scenarios, beyond content-based group recommendation. Here, a primary direction is the evaluation of the proposal in a collaborative filtering-based GRS.

6. Conclusions

This paper presents a novel framework for content-based group recommendation, which main feature is the dynamic selection of the most appropriate aggregation function, for the recommendation aggregation step.

Specifically, it is focused on proposing the use of four attributes for characterizing groups in content-based GRS. Such attributes are the minimum correlation between any pair of group members, the amount of ratings across the group, the amount of co-rated items by all the users, and the rating average of the group. Specifically the proposal is focused on building a fuzzy decision tree that helps to match such attributes of a specific group, with the best aggregation function (average or minimum), that can be use for such group in the individual recommendation aggregation step for improving the recommendation performance.

The proposal is evaluated by an experimental protocol over well-known datasets. The results particularly show that it is able to outperform the baseline for most of all scenarios in the Movielens and HetRec datasets. Furthermore, it was also developed an exploratory analysis of the values of such attributes in all the groups used in the experiments, showing different imbalance degrees that could affect the application of the proposal in a higher or lesser extent.

From a practical viewpoint, our work provides a methodology initially presented for the content-based group recommendation context but that can be also used in other GRSs, in order to guarantee a more dynamic construction of the recommendation architecture, that could result in an improvement of the recommendation accuracy. Furthermore, the nature of the proposal would allow its use in a higher dimension and dense scenarios, considering that its phases tend to have a linear dependency on the dimension of the data, and that most of such phases can be executed in an offline mode for saving computational cost.

The next future work to be developed, already pointed out in the previous section, comprises some goals such as the use of feature engineering approaches for a better characterization of groups, the use of more sophisticate membership functions that represent better the nature of the data, and the use of other supervised classifiers beyond the fuzzy decision tree.

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.

Acknowledgements

This work was supported by the Deanship of Scientific Research (DSR), King Abdulaziz University, Jeddah, under Grant Kep-15-611-42.

References

[1] G. Adomavicius, A.T. Tuzhilin, Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions, IEEE Trans. Knowl. Data Eng. 17 (2005) 734–749.

[2] A. Aizawa, An information-theoretic perspective of tf-idf measures, Inf. Process. Manag. 39 (2003) 45–65.

[3] A.B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, et al., Explainable artificial intelligence (xai): concepts, taxonomies, opportunities and challenges toward responsible ai, Inf. Fusion 58 (2020) 82–115.

[4] L. Baltrunas, M. Kaminskas, B. Ludwig, O. Moling, F. Ricci, A. Aydin, K.-H. Lüke, R. Schwaiger, InCarMusic: Context-Aware Music Recommendations in a Car E-Commerce and Web Technologies, chapter 8, Lecture Notes in Business Information Processing, vol. 85, Springer, Berlin Heidelberg, 2011, pp. 89–100.
[5] L. Baltrunas, T. Makcinskas, F. Ricci, Group recommendations with rank aggregation and collaborative filtering, in: Proceedings of the 4th ACM Conference on Recommender Systems RecSys '10, ACM, New York, NY, USA, 2010, pp. 119–126.
[6] L. Boratto, S. Carta, G. Fenu, Discovery and representation of the preferences of automatically detected groups: exploiting the link between group modeling and clustering, Future Gener. Comput. Syst. 24 (2015) 833–848.
[7] I. Cantador, P. Brusilovsky, T. Kuflik, 2nd workshop on information heterogeneity and fusion in recommender systems (hetrec 2011), in: Proceedings of the 5th ACM Conference on Recommender Systems RecSys 2011, ACM, New York, NY, USA, 2011.
[8] J. Castro, M.J. Barranco, R.M. Rodríguez, L. Martínez, Group recommendations based on hesitant fuzzy sets, Int. J. Intell. Syst. 33 (2018) 2058–2077.
[9] J. Castro, J. Lu, G. Zhang, Y. Dong, L. Martínez, Opinion dynamics-based group recommender systems, IEEE Trans. Syst. Man Cybern. Syst. 48 (2018) 2394–2406.
[10] J. Castro, F.J. Quesada, I. Palomares, L. Martínez, A consensus-driven group recommender system, Int. J. Intell. Syst. 30 (2015) 887–906.
[11] J. Castro, R.M. Rodríguez, M.J. Barranco, Weighting of features in content-based filtering with entropy and dependence measures, Int. J. Comput. Intell. Syst. 7 (2014) 80–89.
[12] J. Castro, R. Yera, A.A. Alzahrani, P. Sanchez, M. Barranco, L. Martínez, A big data semantic driven context aware recommendation method for question-answer items, IEEE Access 7 (2019) 182664–182678.
[13] J. Castro, R. Yera, L. Martínez, An empirical study of natural noise management in group recommendation systems, Decis. Support Syst. 94 (2017) 1–11.
[14] J. Castro, R. Yera, L. Martínez, A fuzzy approach for natural noise management in group recommender systems, Expert Syst. Appl. 94 (2018) 237–249.
[15] Z. Cui, X. Xu, X. Fei, X. Cai, Y. Cao, W. Zhang, J. Chen, Personalized recommendation system based on collaborative filtering for iot scenarios, IEEE Trans. Serv. Comput. 13 (2020) 685–695.
[16] S. Dara, C. Chowdary, C. Kumar, A survey on group recommender systems, J. Intell. Inf. Syst. 54 (2020) 271–295.
[17] T. De Pessemier, J. Dhondt, K. Vanhecke, L. Martens, Travelwithfriends: a hybrid group recommender system for travel destinations, in: Workshop on Tourism Recommender Systems (touRS15), in Conjunction with the 9th ACM Conference on Recommender Systems (recsys 2015), 2015, pp. 51–60.
[18] T. De Pessemier, J. Dhondt, K. Vanhecke, L. Martens, Travelwithfriends: a hybrid group recommender system for travel destinations, in: Workshop on Tourism Recommender Systems (touRS15), in Conjunction with the 9th ACM Conference on Recommender Systems (recsys 2015), 2015, pp. 51–60.
[19] T. De Pessemier, S. Dooms, L. Martens, Comparison of group recommendation algorithms, Multimed. Tools Appl. 72 (2014) 2497–2541.
[20] D. Dubois, H. Prade, Operations on fuzzy numbers, Int. J. Syst. Sci. 9 (1978) 613–626.
[21] M.D. Ekstrand, J.T. Riedl, J.A. Konstan, Collaborative filtering recommender systems, Found. Trends Hum.-Comput. Interact. 4 (2011) 81–173.
[22] A. Felfernig, L. Boratto, M. Stettinger, M. Tkalčič, Group Recommender Systems: An Introduction, Springer, 2018.
[23] B.S. Francesco Ricci, Lior Rokach (Eds.), Recommender Systems Handbook, 2nd ed., Springer US, 2015.
[24] M. de Gemmis, P. Lops, C. Musto, F. Narducci, G. Semeraro, Semantics-aware content-based recommender systems, in: F. Ricci, L. Rokach, B. Shapira (Eds.), Recommender Systems Handbook, Springer US, 2015, pp. 119–159.
[25] A. Gunawardana, G. Shani, A survey of accuracy evaluation metrics of recommendation tasks, J. Mach. Learn. Res. 10 (2009) 2935–2962.
[26] F.M. Harper, J.A. Konstan, The movielens datasets: history and context, ACM Trans. Interact. Intell. Syst. 5 (2015) 19.
[27] F.O. Isinkaye, Matrix factorization in recommender systems: algorithms, applications, and peculiar challenges, IETE J. Res. (2021) 1–14.
[28] P. Jankiewicz, L. Kyrashchuk, P. Sienkowski, M. Wójcik, Boosting algorithms for a session-based, context-aware recommender system in an online travel domain, in: Proceedings of the Workshop on ACM Recommender Systems Challenge, 2019, pp. 1–5.
[29] A. Janusz, S. Stawicki, M. Drewniak, K. Ciebiera, D. Ślęzak, K. Stencel, How to match jobs and candidates-a recruitment support system based on feature engineering and advanced analytics, in: International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, Springer, 2018, pp. 503–514.
[30] S. Kantarci, E. Nasibov, A fuzzy id3 induction for linguistic data sets, Int. J. Intell. Syst. 33 (2018) 858–878.
[31] O. Kaššák, M. Kompan, M. Bieliková, Personalized hybrid recommendation for group of users: top-n multimedia recommender, Inf. Process. Manag. 52 (2016) 459–477.
[32] Y. Koren, R. Bell, C. Volinsky, Matrix factorization techniques for recommender systems, Computer 42 (2009) 30–37.
[33] P. Lops, M. Gemmis, G. Semeraro, Content-based recommender systems: state of the art and trends, in: Recommender Systems Handbook, Springer, US, 2011, pp. 73–105, chapter 3.
[34] T.N. Nguyen, F. Ricci, A chat-based group recommender system for tourism, Inf. Technol. Tour. 18 (2018) 5–28.
[35] M. Nilashi, O. bin Ibrahim, N. Ithnin, N.H. Sarmin, A multi-criteria collaborative filtering recommender system for the tourism domain using expectation maximization (em) and pca–anfis, Electron. Commer. Res. Appl. 14 (2015) 542–562.
[36] C. Park, D. Kim, J. Oh, H. Yu, Using user trust network to improve top-k recommendation, Inf. Sci. 374 (2016) 100–114.
[37] M. Pazzani, D. Billsus, Content-based recommendation systems, in: The Adaptive Web, vol. 4321, 2007, pp. 325–341.
[38] W. Pedrycz, Fuzzy Control and Fuzzy Systems, Research Studies Press Ltd., 1993.
[39] M. Pera, Y. Ng, A group recommender for movies based on content similarity and popularity, Inf. Process. Manag. 49 (2013) 673–687.
[40] M. Pera, Y.K. Ng, Automating readers advisory to make book recommendations for k-12 readers, in: Proceedings of the ACM Conference on Recommender Systems RecSys '14, ACM, 2014, pp. 9–16.
[41] Y. Pérez-Almaguer, R. Yera, A.A. Alzahrani, L. Martínez, Content-based group recommender systems: a general taxonomy and further improvements, Expert Syst. Appl. 184 (2021) 115444.
[42] L. Quijano-Sanchez, J.A. Recio-Garcia, B. Diaz-Agudo, An architecture and functional description to integrate social behaviour knowledge into group recommender systems, Appl. Intell. 40 (2014) 732–748.
[43] F. Ricci, L. Rokach, B. Shapira, Recommender systems handbook, in: F. Ricci, L. Rokach, B. Shapira, P.B. Kantor (Eds.), Recommender Systems Handbook, Springer, 2011, pp. 1–35, chapter 1.
[44] Y.-D. Seo, Y.-G. Kim, E. Lee, K.-S. Seol, D.-K. Baik, An enhanced aggregation method considering deviations for a group recommendation, Expert Syst. Appl. 93 (2018) 299–312.
[45] L.H. Son, Hu-fcf++: a novel hybrid method for the new user cold-start problem in recommender systems, Eng. Appl. Artif. Intell. 41 (2015) 207–222.
[46] M. Umanol, H. Okamoto, I. Hatono, H. Tamura, F. Kawachi, S. Umedzu, J. Kinoshita, Fuzzy decision trees by fuzzy id3 algorithm and its application to diagnosis systems, in: Proceedings of 1994 IEEE 3rd International Fuzzy Systems Conference, IEEE, 1994, pp. 2113–2118.
[47] J. Wang, Y. Jiang, J. Sun, Y. Liu, X. Liu, Group recommendation based on a bidirectional tensor factorization model, World Wide Web 21 (2018) 961–984.
[48] J. Xie, S. Leishman, L. Tian, D. Lisuk, S. Koo, M. Blume, Feature engineering in user's music preference prediction, in: Proceedings of KDD Cup 2011, PMLR, 2012, pp. 183–197.
[49] E. Yalcin, F. Ismailoglu, A. Bilge, An entropy empowered hybridized aggregation technique for group recommender systems, Expert Syst. Appl. 166 (2021) 114111.
[50] R. Yera, A.A. Alzahrani, L. Martínez, A food recommender system considering nutritional information and user preferences, IEEE Access 7 (2019) 96695–96711.

[51] R. Yera, J. Castro, L. Martínez, A fuzzy model for managing natural noise in recommender systems, Appl. Soft Comput. 40 (2016) 187–198.
[52] R. Yera, L. Martínez, Fuzzy tools in recommender systems: a survey, Int. J. Comput. Intell. Syst. 10 (2017) 776–803.
[53] R. Yera, L. Martínez, A recommendation approach for programming online judges supported by data preprocessing techniques, Appl. Intell. 47 (2017) 277–290.
[54] L.A. Zadeh, Fuzzy sets, Inf. Control 8 (1965) 338–353.
[55] Z. Zeeshan, U.A. Bhatti, W.H. Memon, S. Ali, S.A. Nawaz, M.M. Nizamani, A. Mehmood, M.A. Bhatti, M.U. Shoukat, et al., Feature-based multi-criteria recommendation system using a weighted approach with ranking correlation, Intell. Data Anal. 25 (2021) 1013–1029.
[56] Y. Zhang, Grorec: a group-centric intelligent recommender system integrating social, mobile and big data technologies, IEEE Trans. Serv. Comput. 9 (2016) 786–795.