Expert Systems With Applications 94 (2018) 237–249

Contents lists available at ScienceDirect

Expert Systems With Applications

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

A fuzzy approach for natural noise management in group recommender systems

Jorge Castro[a,b,*], Raciel Yera[c], Luis Martínez[d,e]

[a] Department of Computer Science and Artificial Intelligence, University of Granada, Calle Periodista Daniel Saucedo Aranda s/n, Granada 18071, Spain
[b] School of Software, University of Technology Sydney, PO Box 123, Broadway, Ultimo, NSW 2007, Australia
[c] University of Ciego de Ávila, Carretera a Morón Km. 9 1/2, Ciego de Ávila, Cuba
[d] Computer Science Department, University of Jaén, Campus Las Lagunillas s/n, Jaén 23008, Spain
[e] School of Management, Wuhan University of Technology, Wuhan 430070, China

ARTICLE INFO

Article history:
Received 4 May 2017
Revised 27 October 2017
Accepted 31 October 2017
Available online 8 November 2017

Keywords:
Natural noise
Group recommender systems
Collaborative filtering
Fuzzy logic
Computing with words

ABSTRACT

Information filtering is a key task in scenarios with information overload. Group Recommender Systems (GRSs) filter content regarding groups of users preferences and needs. Both the recommendation method and the available data influence recommendation quality. Most researchers improved group recommendations through the proposal of new algorithms. However, it has been pointed out that the ratings are not always right because users can introduce noise due to factors such as context of rating or user's errors. This introduction of errors without malicious intentions is named natural noise, and it biases the recommendation. Researchers explored natural noise management in individual recommendation, but few explored it in GRSs. The latter ones apply crisp techniques, which results in a rigid management. In this work, we propose Natural Noise Management for Groups based on Fuzzy Tools (NNMG-FT). NNMG-FT flexibilises the detection and correction of the natural noise to perform a better removal of natural noise influence in the recommendation, hence, the recommendations of a latter GRS are then improved.

1. Introduction

The Web allows people accessing to a huge amount of information. However, the users skills to cope with all the available information are limited, which leads to select suboptimal alternatives. This problem is known as information overload. Recommender Systems (RSs) are tools to help individuals to overcome such information overload problem personalizing access to information (Adomavicius & Tuzhilin, 2005; Ekstrand, Riedl, & Konstan, 2011). However, some items tend to be consumed by groups of users, such as tourist attractions (Garcia, Pajares, Sebastia, & Onaindia, 2012) or television programmes (Said, Berkovsky, & De Luca, 2011). With this purpose in mind, Group Recommender Systems (GRSs) (Masthoff, 2015) help groups of users to find suitable items according to their preferences and needs.

Several techniques have been used to improve individual recommendation, such as neighborhood-based collaborative filtering (Sarwar, Karypis, Konstan, & Riedl, 2001), matrix factorisation (Koren, Bell, & Volinsky, 2009), or approaches that consider temporal dynamics (Koren, 2010; Rafailidis, Kefalas, & Manolopoulos, 2017). In the case of group recommendation, there are approaches to aggregate individual information (Masthoff, 2015), to consider consensus among members (Castro, Quesada, Palomares, & Martínez, 2015), or matrix factorisation models for groups (Ortega, Hernando, Bobadilla, & Kang, 2016).

A decade ago, it was pointed out that explicitly stated user preferences may not be error free (O'Mahony, Hurley, & Silvestre, 2006). More recently, other recent works (Bellogín, Said, & de Vries, 2014; Centeno, Hermoso, & Fasli, 2015; Guo & Dunson, 2015; Zhang, Zhao, & Lui, 2017) have also pointed out that a person's ratings are noisy, inconsistent, and biased. Li, Chen, Zhu, and Zhang (2013) determined that too many noisy ratings can distort users' preference profiles, which result in *unlike-minded* neighbors that imply a quality loss in recommendations. Kluver, Nguyen, Ekstrand, Sen, and Riedl (2012) have also suggested that user ratings are imperfect and noisy, and such noise limits the predictive power of any RS.

Therefore, in addition to improving recommendations through new recommendation approaches, researchers should also focus on improving the quality of the rating database (Amatriain, Pujol, Tintarev, & Oliver, 2009c). In RSs, there are two kinds of noise in the database (O'Mahony et al., 2006): (i) *malicious noise*, that

* Corresponding author at: Department of Computer Science and Artificial Intelligence, University of Granada, Calle Periodista Daniel Saucedo Aranda s/n, Granada 18071, Spain.
E-mail addresses: jcastro@decsai.ugr.es (J. Castro), ryera@unica.cu (R. Yera), martin@ujaen.es (L. Martínez).

https://doi.org/10.1016/j.eswa.2017.10.060

consists of erroneous data deliberately inserted in the system to influence recommendations, and (ii) *natural noise*, that appears when users unpurposely introduce erroneous data due to human errors or external factors during the rating process. This paper focuses on the latter.

Natural noise biases recommendations, therefore, its management is a key factor to improve them. There are several Natural Noise Management (NNM) approaches for individual RSs databases. While some NNM approaches need additional information (Amatriain, Lathia, Pujol, Kwak, & Oliver, 2009a; Pham & Jung, 2013), others detect and correct the natural noise using information already contained in the database (Yera, Castro, & Martínez, 2016; Yera Toledo, Caballero Mota, & Martínez, 2015).

GRSs also rely on databases with explicit users' preferences (Masthoff, 2015), therefore, they are affected by natural noise. Castro, Yera, and Martínez (2017) propose a NNM approach for GRSs to manage ratings and noise using crisp values. This is the only work focused on NNM in GRSs. However, the crisp management is not either flexible or robust enough to deal with the uncertainty and vagueness of both the ratings and the NNM, which makes it necessary to develop new proposals with this regard.

In order to manage such uncertainty and vagueness in RSs contexts, the use of fuzzy tools has been considered for several years. A recent survey paper (Yera & Martínez, 2017) has shown that some traditional fuzzy tools have been successfully used for a more flexible and accurate information processing in RSs. However, it also shows that there are several research gaps related to the necessity of new fuzzy approaches focused on the use of emergent information sources and concentrated in new research trends in RSs. Specifically, the natural noise management (Martínez, Castro, & Yera, 2016) is one of such research trends. Our purpose is to study the natural noise management in group recommendation with fuzzy tools.

Therefore, in this work we propose Natural Noise Management for Groups based on Fuzzy Tools (NNMG-FT) to improve the rating database removing the natural noise. NNMG-FT applies three steps of management: fuzzy profiling, global noise management and local noise management. Both global natural noise management step and local noise management step are divided into two substeps: noise detection and noise correction. Both sub-steps apply fuzzy tools. In the noise detection, fuzzy tools allow to make a flexible classification of the ratings into noisy or not noisy. In the noise correction, this flexible classification is used to correct noisy ratings applying a soft modification of the value regarding its noise degree. The main advantages of NNMG-FT are: flexibility, robustness and consideration of group information in the NNM. A case study was performed to show the validity of NNMG-FT.

In short, the main contributions of this paper consist of:

- Design an improved profiling that manages uncertainty and vagueness of the ratings through the application of fuzzy tools in the profiling of ratings, users, and items.
- Design an adequate representation of the noise management process that improves the flexibility and robustness of the noise detection and noise correction.
- Propose a NNM approach for GRSs that hybridizes several steps of noise detection and correction based on the information level from the viewpoints of both the whole ratings database and the groups ratings.
- Validate the proposal through comparison with previous ones with similar purpose.

The remainder of this paper is structured as follows. First, Section 2 presents the related works for the current research. Section 3 details NNMG-FT, our proposal for NNM in group recommendation. Section 4 shows the case study done to validate NNMG-FT performance. Finally, Section 5 concludes the work.

2. Related works

In this section we revise different concepts about natural noise management in recommender systems, GRSs, and fuzzy sets, that are used in our NNM approach for GRSs.

2.1. Natural noise management

The existence of underlying noise in users' preferences in RSs and its negative effect have been referred for several years. In this way, an influential paper presented by Herlocker, Konstan, Terveen, and Riedl (2004) pointed out that, although an important amount of advanced algorithms were developed for improving RSs accuracy, the mean absolute error tends to yield around a constant magnitude. They speculated then that such algorithms could be reaching some *magic barrier* where natural variability in ratings may prevent researches from getting much more accurate results. The existence of such *magic barrier* has been confirmed by further investigations in the last few years (Bellogín et al., 2014; Said, Jain, Narr, & Plumbaum, 2012), which have been focused on its characterisation and estimation.

Additionally, the underlying noise in users' preferences began to be referred as natural noise. Formally, natural noise term was first coined by O'Mahony et al. (2006) as those inconsistencies introduced in recommender systems databases due to the imperfect users behaviour when they rate the reviewed or purchased products, without a premeditated malicious intention. It is produced by the influence of external factors in the rating process, such as human errors or rating in different contexts. Natural noise influences the quality of user ratings, and researchers have determined that this influence results in poor recommendations (Amatriain, Pujol, & Oliver, 2009b; Amatriain et al., 2009c). Therefore, an adequate Natural Noise Management (NNM) is key to improve recommendations.

Researchers have explored NNM for individual RSs, which is applied as a preprocessing stage done over the ratings database to reduce the impact of noisy information. Some techniques remove noisy information from the rating database, such as O'Mahony et al. (2006), which deletes both malicious and natural noisy ratings, or Li et al. (2013), which eliminates noisy but non malicious users. These works use the information already contained in the ratings database. However, they overlook important information from the dataset.

There are works that rely on additional information to correct natural noisy ratings. Amatriain et al. (2009c) propose the mining and usage of a curated dataset with information provided by experts to reduce noise. Pham and Jung (2013) uses item attributes to build user models and correct ratings not matching the model, which is built using information of other users identified as experts. More recently, Bellogín et al. (2014) use item attributes for measuring user coherence in recommender systems databases, showing that the recommendation performance is improved when less coherent users are discarded. Later, Yu, Lin, and Yao (2016) propose a correction approach for ratings associated to such less coherent users. Additionally, Saia, Boratto, and Carta (2016) have presented an approach for removing incoherent items from a user profile, using semantic information. These approaches need additional information to correct the noisy ratings, which may not be feasible to obtain in certain domains. Recent proposals also focus on the detection and correction of natural noisy ratings using information contained in the original database. Some of these proposals use contradiction-based approaches (Yera Toledo et al., 2015) or fuzzy tools (Yera et al., 2016).

On the other hand, in RSs context there are items that, because of their social features, tend to be consumed by groups, such as tour packages for groups of tourists (Ardissono, Goy, Petrone,

Table 1
Research works focused on natural noise management.

		Target	
		Individual	Group
NNM	Crisp	O'Mahony et al. (2006), Amatriain et al. (2009b, 2009c), Li et al. (2013), Pham and Jung (2013), Bellogín et al. (2014), Saia et al. (2016), Yu et al. (2016), Yera Toledo et al. (2015)	Castro et al. (2017)
	Fuzzy	Yera et al. (2016)	This contribution

Segnan, & Torasso, 2003), playlists for groups of listeners (Crossen, Budzik, & Hammond, 2002), or healthy food for groups of family members, friends or colleagues (Trang Tran, Atas, Felfernig, & Stettinger, 2017). In the social items scenario, Group Recommender Systems (GRSs, see next Section 2.2) have emerged as an effective solution for providing recommendations to groups of people (Castro et al., 2015; De Pessemier, Dooms, & Martens, 2014). Nevertheless, the NNM in GRSs has been an unexplored research area, regarding that most of the revised works focus on managing natural noise in individual RSs. The only reported research considering NNM in GRSs has been recently presented by Castro et al. (2017). Although the approaches proposed in such work introduced improvements in the recommendation accuracy, it also has some important shortcomings. Specifically, it does not consider the uncertainty and vagueness associated to the users preferences, which have been considered in NNM process for individual RSs (Yera et al., 2016). The latter NNM approach proved to improve recommendation accuracy in comparison to previous crisp models for individual RSs (Li et al., 2013; Yera Toledo et al., 2015).

Table 1 summarises the referred works and classifies them according to the recommendation context, individual or group, and to the techniques used for the NNM. Such table suggests that the NNM in groups using fuzzy techniques is still an area to explore. In this work, we aim to fill this gap with a new approach that, in contrast to the previous ones, performs an intensive use of fuzzy techniques both in the preferences of the active group and in the preferences of all available users, to perform a flexible and robust NNM. These features enhance the quality of the corrected users' preferences and, therefore, improve the recommendations of the associated GRS.

The *magic barrier* in accuracy (Bellogín et al., 2014) is not only caused by natural noise. Recommendation accuracy is also limited by temporal dynamics (Zhang, Wang, Yu, Sun, & Lim, 2014), which study the evolution of users preferences across time due to changes in users taste. Different proposals have studied this important issue in RSs, such as time-based collective factorisation (Vaca, Mantrach, Jaimes, & Saerens, 2014), temporal matrix factorisation (Zhang et al., 2014) or combination of multi-modal and temporal information (Rafailidis et al., 2017). Although temporal dynamics is related to our research, it is based on another view of the experts' preferences and it is not simultaneously considered with natural noise in this paper to avoid digressing and mix up its goal.

2.2. Group recommender systems

GRSs provide groups of users with group personalised access to information (Masthoff, 2015). The group recommendation problem has been formalised, using the notation shown in Table 2, as finding the item, or set of items, that maximise the aggregated prediction for the target group:

$$Recommendation(G_a, I) = arg\max_{i\in I} Prediction(G_a, i) \quad (1)$$

Table 2
Notation used for group recommender systems.

Symbol	Description
$U = \{u_1, \ldots, u_m\}$	Set of all users
$I = \{i_1, \ldots, i_n\}$	Set of all items
$R \subseteq U \times I$	Set of known ratings.
$r_{ui} \in R$	Rating that user u gave about item i
$[r_{\min}, r_{\max}]$	Rating domain of the dataset given between $r_{\min}$ and $r_{\max}$
$R_u \subseteq R$	Set of ratings given by user u.
$R_i \subseteq R$	Set of ratings for item i.
$G_a = \{m_1, \ldots, m_g\} \subseteq U$	Target group with g members
$R_i^{G_a} \subseteq R_i$	Ratings that members of group G_a provided for item i.

Table 3
Notation used for fuzzy sets.

Symbol	Description
$x \in X$	Value given in the X domain.
$\tilde{A}$	Fuzzy set defined over the values of X domain.
$\mu_{\tilde{A}}(x)$	Membership function that gives membership of value x to $\tilde{A}$.
$\tilde{A}_\alpha \subseteq X$	Alpha-cut of the fuzzy set $\tilde{A}$
$f(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x))$	t-norm (intersection) of fuzzy sets
$g(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x))$	t-conorm (union) of fuzzy sets

where G_a is the target group, I is the set of items in the database and $Prediction(G_a, i)$ predicts the rating that group G_a would give to item i and is given in the $[r_{\min}, r_{\max}]$ domain.

Among the various ways to compute the prediction for the group, the most successful ones are based on aggregating individual information to recommend:

- Rating aggregation (Kagita, Pujari, & Padmanabhan, 2015): The rating profiles of all members are aggregated into a single rating profile that represents the group preferences. This group profile is then used in an individual recommender system to recommend.
- Recommendation aggregation (O'Connor, Cosley, Konstan, & Riedl, 2001): An individual RS produces recommendations for each member, which are later aggregated into a single recommendation for the group.

It is worth to mention that, within these approaches, various aggregation strategies can be applied. Most successful strategies are average (Yu, Zhou, Hao, & Gu, 2006), least misery (O'Connor et al., 2001) and multiplicative (McCarthy & Anagnost, 1998).

When ratings are explicitly elicited by users, they can be affected by natural noise (O'Mahony et al., 2006). Moreover, some aggregations are more robust to noise, such as average, but others are more sensitive, such as least misery and multiplicative strategies. Therefore, NNM is key to improve recommendations in this scenario, as it was pointed out in Section 2.1.

2.3. Fuzzy sets

Previous NNM approaches for GRSs are too rigid, as it was pointed out in Section 2.1. This results in a limited noise management that needs to be improved through the flexibilisation of the NNM. With such aim in mind, we use fuzzy tools (Table 3).

For traditional sets, a given element either belongs or does not belong to the set. However, some concepts cannot be precisely defined, such as *old* person, *comfortable* seat or *good* taste. Fuzzy sets (Yager & Zadeh, 2012; Zadeh, 2012) extend the definition of sets to manage uncertain or vague information through the inclusion of a degree of membership of elements. The membership establishes a correspondence between the elements of the universe of discourse

X into the interval [0, 1], and it can be given as a membership function:

$$\mu_{\tilde{A}} : X \rightarrow [0, 1]$$

Specifically, a fuzzy set $\tilde{A}$ defined over the discrete domain X is represented by the set of pairs of the elements $x \in X$ and their corresponding memberships:

$$\tilde{A} = \{ \ (x, \mu_{\tilde{A}}(x)) \quad / \quad x \in X, \quad \mu_{\tilde{A}}(x) \in [0, 1] \ \} \tag{2}$$

Additionally, the α-cut of $\tilde{A}$ is defined as a classic subset of elements in X, whose membership function takes a greater or equal value to any specific α value of that universe of discourse that complies with:

$$\tilde{A}_\alpha = \{x \in \mathbb{R} \mid \mu_{\tilde{A}}(x) \geq \alpha\}. \tag{3}$$

With this definition of fuzzy set, the notions of intersection and union over traditional sets are also extended to be defined for fuzzy scenarios (Zadeh, 1965; Zimmermann, 2001). This way, the intersection of two fuzzy sets $\tilde{A}$ and $\tilde{B}$ is a fuzzy set $\tilde{C}$ (see Eq. (4)) where f is a t-norm. Additionally, the union between $\tilde{A}$ and $\tilde{B}$ is a fuzzy set $\tilde{D}$, where g is a t-conorm (see Eq. (5)):

$$\mu_{\tilde{C}}(x) = f(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)), \qquad x \in X \tag{4}$$

$$\mu_{\tilde{D}}(x) = g(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)), \qquad x \in X \tag{5}$$

Both T_{norms} and $T_{conorms}$ are operators that establish generic models for the operations of union and intersection, which must comply with certain basic properties (Schweizer & Sklar, 2011): commutative, monotonicity, associative, and border conditions. The most common definition of these operators are the $t-norm$ of the Minimum and the $t-conorm$ of the Maximum.

In addition to the mentioned basic foundations in fuzzy set theory, it is worthy to mention the use of the fuzzy linguistic approach (Herrera & Martinez, 2000; Rodríguez, Labella, & Martínez, 2016; Rodríguez & Martínez, 2013) for modelling uncertainty and vagueness through linguistic variables. Such an approach requires the selection of appropriate linguistic descriptors for the corresponding term set, the syntax and the semantics; where the semantics is represented by using fuzzy memberships functions.

In this work, the concepts introduced in this section are applied in the natural noise preprocessing to build profiles for users, items and ratings, and to characterise the noise degree of a given rating in a flexible and robust way.

3. Natural noise management for groups based on fuzzy tools

Previous approach for GRS with NNM (Castro et al., 2017) does not manage the inherent uncertainty and vagueness of the noise. We aim to fill this gap by proposing an approach for NNM for Groups based on Fuzzy Tools (NNMG-FT).

NNMG-FT analyses the rating database to detect noisy ratings and correct them to reduce their impact in a latter GRS. NNMG-FT has three main phases, as Fig. 1 shows:

1. Fuzzy profiling: Generates a representation for users, items and ratings to characterise them and facilitate their analysis in following phases.
2. Global noise management: Based upon these profiles, it manages the natural noise of the rating database in a global level.
3. Local noise management: It uses the corrected information obtained in the previous step, to perform a noise management focused on the group ratings.

The fuzzy profiling generates profiles for users, items and ratings to be analysed. These profiles are used to perform an initial NNM that results in a de-noised ratings dataset. After that, the proposal performs a NNM over the de-noised ratings dataset focused on the group ratings to further refine it. As a result, a rating dataset without natural noise is obtained, which can later be used by a GRS to recommend. A detailed description of these phases is presented in the remaining of this section using the notation in Table 4.

Table 4
Notation used for the proposal.

Symbol	Description
$p_{r_{ui}}$	Fuzzy rating profile of rating r_{ui}.
p_u	Fuzzy user profile of user u.
p_i	Fuzzy item profile of item i.
$p_i^{G_a}$	Fuzzy item profile restricted to ratings given by G_a members.
$p_{r_{ui}}, p_u^*, p_i^*, p_i^{*G_a}$	Modified fuzzy profiles.
n_{ui}	Rating prediction for user u and item i
$r_{ui}^* \in R^*$	Rating for user u and item i after the global noise management phase
$r_{ui}^{**} \in R^{**}$	Rating after the local noise management phase

3.1. Fuzzy profiling

With NNMG-FT, we aim to reduce the natural noise analysing rating tendencies of users, items and ratings. With this aim, this phase generates fuzzy profiles for them in order to facilitate their analysis in following phases. The fuzzy profiles contain the characterisation of the rating tendency in the linguistic variable, which has the linguistic terms *low, medium* or *high*. The input of this phase is the rating dataset, and the outputs are rating, user, item, and group-based item fuzzy profiles.

First, we define the fuzzy profile for single ratings. Using such rating fuzzy profiles, we define users and items fuzzy profiles. We consider the n rating domain to characterize ratings across their universe of discourse. With this regard, we propose the use of the membership functions presented in Fig. 2, whose definition has been determined empirically (Yera et al., 2016). Therefore, a rating fuzzy profile is a 3-tuple (Eq. (6)), which contains the membership degree to the fuzzy sets *low, medium*, and *high*.

$$p_{r_{ui}} = (\mu_{low}(r_{ui}), \mu_{medium}(r_{ui}), \mu_{high}(r_{ui})) \tag{6}$$

With this fuzzy profile for ratings we can build fuzzy profiles for users and for items, which are noted as p_u and p_i, respectively. Similarly to rating fuzzy profiles, they are composed of 3-tuples (Eqs. (7) and (8)). These fuzzy profiles are built averaging the membership degrees of their associated ratings to capture the rating tendency of users and items, respectively.

$$\begin{aligned} p_u &= \left(p_{u_{low}}, p_{u_{medium}}, p_{u_{high}}\right) \\ &= \left(\frac{1}{|R_u|}\sum_{r_{ui}\in R_u}\mu_{low}(r_{ui}), \frac{1}{|R_u|}\sum_{r_{ui}\in R_u}\mu_{medium}(r_{ui}), \right. \\ &\qquad \left. \frac{1}{|R_u|}\sum_{r_{ui}\in R_u}\mu_{high}(r_{ui})\right) \end{aligned} \tag{7}$$

$$p_i = \left(p_{i_{low}}, p_{i_{medium}}, p_{i_{high}}\right) \tag{8}$$

$$\begin{aligned} &= \left(\frac{1}{|R_i|}\sum_{r_{ui}\in R_i}\mu_{low}(r_{ui}), \right. \\ &\qquad \left. \frac{1}{|R_i|}\sum_{r_{ui}\in R_i}\mu_{medium}(r_{ui}), \frac{1}{|R_i|}\sum_{r_{ui}\in R_i}\mu_{high}(r_{ui})\right) \end{aligned} \tag{8}$$

In addition to user and item profiles, a *group-based item fuzzy profile* is also defined to characterise items regarding the target group, which will be used in the third phase (Section 3.3). They

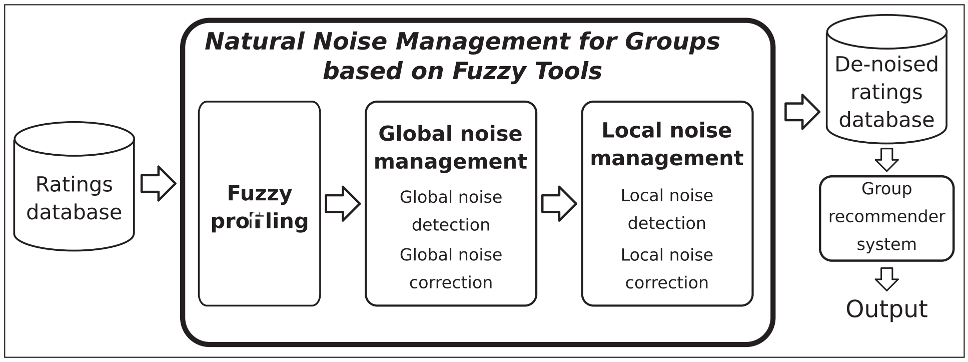


Fig. 1. General scheme of NNMG-FT.

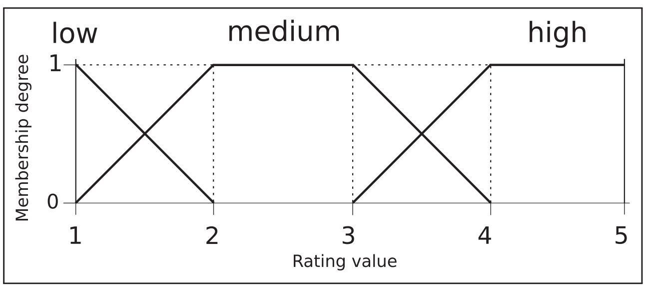


Fig. 2. Membership functions of the ratings domain for the one to five stars domain.

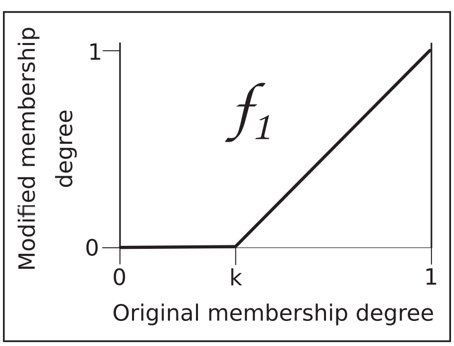


Fig. 3. The fuzzy transformation function.

are built in a similar way to item profiles, but using only ratings of group G_a members, instead of all the available users (Eq. (9)). Therefore, for a given item i only the ratings in $R_i^{G_a}$, which are associated to the corresponding group G_a, are considered for the group-based item profile, instead of all ratings in R_i.

$$p_i^{G_a} = \left(p_{i_{low}}^{G_a}, p_{i_{medium}}^{G_a}, p_{i_{high}}^{G_a}\right) = \left(\frac{1}{|R_i^{G_a}|}\sum_{r_{ui}\in R_i^{G_a}} \mu_{low}(r_{ui}), \frac{1}{|R_i^{G_a}|}\sum_{r_{ui}\in R_i^{G_a}} \mu_{medium}(r_{ui}), \frac{1}{|R_i^{G_a}|}\sum_{r_{ui}\in R_i^{G_a}} \mu_{high}(r_{ui})\right) \quad (9)$$

The fuzzy profiles characterise the rating tendency. In the NNM we aim to detect clear tendencies. With this regard, the fuzzy profiles are modified using a soft modification that can be formulated in various ways depending of the aim. The proposal uses transformation function f_1, presented in Fig. 3, which depends on parameter k that indicates the extent to which unclear tendencies are attenuated. The best value for parameter k is determined through an empirical analysis.

Therefore, Eqs. (10)–(13) present the transformed user, item and rating profiles, which are used in the following phases.

$$p_u^* = \left(f_1(p_{u_{low}}), f_1(p_{u_{medium}}), f_1(p_{u_{high}})\right) \quad (10)$$

$$p_i^* = \left(f_1(p_{i_{low}}), f_1(p_{i_{medium}}), f_1(p_{i_{high}})\right) \quad (11)$$

$$p_i^{*G_a} = \left(f_1(p_{i_{low}}^{G_a}), f_1(p_{i_{medium}}^{G_a}), f_1(p_{i_{high}}^{G_a})\right) \quad (12)$$

$$p_{r_{ui}}^* = \left(f_1(\mu_{low}(r_{ui})), f_1(\mu_{medium}(r_{ui})), f_1(\mu_{high}(r_{ui}))\right) \quad (13)$$

3.2. Global noise management

Once the fuzzy profiles are obtained, the global rating correction phase is performed, which is presented in this section. This phase aims to perform an initial reduction of the natural noise from the viewpoint of the entire ratings database. To do so, this phase is divided into two steps: (a) global noise detection, which detects noisy ratings, and (b) global noise correction, which modifies the noisy ratings value to reduce the natural noise in the ratings database. The inputs of this phase are the rating dataset and the fuzzy profiles obtained in the previous phase. As output, this phase produces a de-noised ratings database, which will be used in the third phase.

3.2.1. Global noise detection

This step develops an exhaustive analysis of each rating to find noisy ones, therefore its inputs are the ratings database and the fuzzy profiles, and its output is a list of detected noisy ratings. Specifically, the aim is to find ratings whose corresponding user and item have consistent rating tendencies and the rating itself is not coherent with them because this situation might indicate that the rating value is noisy. Eq. (14) formalises this strategy to decide whether the current rating r_{ui} is noisy or not using these two conditions.

The first condition evaluates, for a rating r_{ui}, whether its corresponding user fuzzy profile p_u^* and item fuzzy profile p_i^* have similar preference tendencies checking whether they are close enough.

The second condition evaluates whether the rating fuzzy profile $p_{r_{ui}}^*$ is far enough from both the user and the item fuzzy profiles. It means that the rating value does not match its corresponding user and item tendencies. If the rating r_{ui} satisfies both conditions, then it is considered as noisy.

$$\underbrace{(d(p_u^*, p_i^*) < \delta_1}_{first\ condition}\ and\ \underbrace{min(d(p_u^*, p_{r_{ui}}^*),\quad d(p_i^*, p_{r_{ui}}^*)) >= \delta_2)}_{second\ condition} \rightarrow r_{ui}\ is\ noisy \quad (14)$$

Eq. (14) depends on a dissimilarity function d. In this proposal, the Manhattan distance is used as dissimilarity measure (see Eq. (15)) because it reflects the differences between dimensions without giving importance to how these differences are distributed among dimensions, as euclidean distance would do. The best values for δ_1 and δ_2 are determined through an empirical analysis.

$$d(p^*_u, p^*_i) = \sum_s |p^*_{u_s} - p^*_{i_s}| = |p^*_{u_{low}} - p^*_{i_{low}}| + |p^*_{u_{medium}} - p^*_{i_{medium}}| + |p^*_{u_{high}} - p^*_{i_{high}}| \quad (15)$$

3.2.2. Global noise correction

Once the noisy ratings have been identified, this step aims at correcting them. This step receives as input the list of detected noisy ratings, and performs a flexible correction of the noisy ratings whose output is a de-noised ratings database. To do so, noisy ratings are characterised by their noise degree, which controls the extent of the correction. This noisy degree is computed using the fuzzy profiles used in the previous phases.

The formal definition of the noise degree is given by Eq. (16), which is computed considering the minimum dissimilarity between the rating, and user or item fuzzy profiles, i.e., the value of the second condition in Eq. (14). The Manhattan distance between fuzzy profiles is restricted to the interval [1, 2], as it was proved in Yera et al. (2016). Thus, the noise degree is normalised subtracting 1 to the minimum distance.

$$NoiseDegree_{r_{ui}} = min(d(p^*_u, p^*_{r_{ui}}), \quad d(p^*_i, p^*_{r_{ui}})) - 1 \quad (16)$$

The de-noised rating value r^*_{ui} is computed through a convex combination of the rating value r_{ui} and the prediction n_{ui}, which is controlled by the noise degree. Eq. (17) formalises this procedure, which is applied to all noisy ratings.

$$r^*_{ui} = r_{ui} * (1 - NoiseDegree_{r_{ui}}) + n_{ui} * NoiseDegree_{r_{ui}} \quad (17)$$

where n_{ui} is a predicted rating value that is obtained using a collaborative filtering rating prediction approach that considers all the available rating data. We propose to use the user-based collaborative filtering approach (Ning, Desrosiers, & Karypis, 2015), as previous researches used (Li et al., 2013), although other approaches could be used.

$$n_{ui} = Prediction(u, i, R) \in [r_{min}, r_{max}] \quad (18)$$

As a result of this phase, the corrected ratings dataset R^* is obtained, which is used as input in the following phase.

3.3. Local noise management

Once the global noise management phase is completed, the last phase applies a similar NNM focused in the local level, i.e., focused on the group ratings. To do so, this phase takes as inputs the database R^* already de-noised by phase 2 and the corresponding fuzzy profiles. It refines R^* with a NNM focused on the target group ratings, which generates a new ratings database R^{**} as output. This phase is also composed of two steps: (i) local noise detection and (ii) local noise correction.

3.3.1. Local noise detection

In the case of the local noise detection, all the ratings of group G_a members received as input, are checked to verify whether they are natural noisy. Eq. (19) presents the criteria used to detect whether rating r_{ui} is noisy using in this case the group-based item profile (see Eq. (9)). This step produces a list of noisy ratings as output.

$$(d(p^*_u, p^{G_a*}_i) < \delta_1 \text{ and } min(d(p^*_u, p^*_{r_{ui}}), \quad d(p^{G_a*}_i, p^*_{r_{ui}})) >= \delta_2) \rightarrow r_{ui} \text{ is noisy} \quad (19)$$

3.3.2. Local noise correction

The local noise correction step corrects all the ratings identified as noisy in the local noise detection step (input), and produces a de-noised ratings database (output). Here, similarly to the previous phase, the group-based fuzzy item profile is considered. Eq. (20) formalises the calculation of the noise degree regarding the group G_a.

$$NoiseDegree^{G_a}_{r^*_{ui}} = min(d(p^*_u, p^*_{r_{ui}}), \quad d(p^{G_a*}_i, p^*_{r_{ui}})) - 1 \quad (20)$$

The noise correction is performed then over the noisy ratings using Eq. (21). As a result, the noise in the rating database is reduced with a correction adjusted to the target group.

$$r^{**}_{ui} = r^*_{ui} * (1 - NoiseDegree^{G_a}_{r^*_{ui}}) + n^*_{ui} * NoiseDegree^{G_a}_{r^*_{ui}} \quad (21)$$

where n^*_{ui} is a predicted rating value that is obtained from the corrected rating database R^*. Similarly to the computation of n_{ui}, we propose to use the user-based collaborative filtering approach (Ning et al., 2015).

$$n^*_{ui} = Prediction(u, i, R^*) \in [r_{min}, r_{max}] \quad (22)$$

This last step concludes the third phase and, therefore, NNMG-FT. The output is a de-noised rating database R^{**} that can later be used by a GRS to recommend.

4. Case study

We developed a case study to evaluate NNMG-FT. The remaining of this section details the experimental protocol and shows its results.

4.1. Experimental protocol

To evaluate NNMG-FT, we used an experimental procedure based on a popular protocol for group recommendation (De Pessemier et al., 2014). Such experimental procedure is composed of the following steps:

- Partition the rating dataset in training and test sets randomly.
- Generate the groups randomly.
- Apply the proposed NNM approach to the training set.
- Recommend to each group regarding the data in the training set and the group recommendation algorithm.
- Evaluate the recommendations using the test set.

Within this experimental protocol, the Mean Absolute Error has been considered as the evaluation metric. Specifically, such protocol was repeated 20 times and the values obtained in each execution were averaged. In each execution, 50 different random groups were generated to evaluate their recommendations. Groups sizes 5, 10, and 15 were considered in the case study.

In this evaluation, three NNM approaches were compared: (i) Base, (ii) NNMG-Crisp, crisp NNM for groups (Castro et al., 2017), and (iii) the current proposal NNMG-FT, NNM for groups based on fuzzy tools. To quantify the effect of each NNM approach, we measured the MAE of various GRSs with de-noised ratings databases using these NNM approaches. To perform a comprehensive analysis, we evaluated them with GRSs based on rating and recommendation aggregation approaches (De Pessemier et al., 2014). Various aggregation strategies can be applied within each of these aggregation approaches. Specifically, average and least misery strategies were used in the case study.

These GRSs rely on an individual RS to recommend. All evaluated GRSs use the item-based collaborative filtering approach (Sarwar et al., 2001). This approach has been a very popular collaborative filtering method whose importance is high in the RSs field due to its simplicity, effectiveness and scalability (Adomavicius & Tuzhilin, 2005; Ekstrand et al., 2011).

Table 5
NNMG-FT parameter optimisation. Optimisation of k using MAE.

Aggregation approach	Dataset	Aggregation strategy	Group size	k 0.35	0.50	0.75
Rating aggregation	MovieLens 100k	Mean	5	**0.8364**	0.8503	0.8652
			10	**0.8590**	0.8684	0.8883
			15	**0.8686**	0.8898	0.9093
		Min	5	**0.8545**	0.8615	0.8864
			10	**0.9177**	0.9301	0.9544
			15	**0.9805**	0.9962	1.0247
	Netflix Tiny	Mean	5	**0.8272**	0.8416	0.8612
			10	**0.8517**	0.8685	0.8857
			15	**0.8566**	0.8698	0.8907
		Min	5	**0.8555**	0.8730	0.8891
			10	**0.9167**	0.9327	0.9520
			15	**0.9595**	0.9703	0.9891
Recomm. aggregation	MovieLens 100k	Mean	5	**0.8335**	0.8497	0.8791
			10	**0.8590**	0.8726	0.8987
			15	**0.8802**	0.8980	0.9110
		Min	5	**0.9855**	1.0115	1.0900
			10	**1.1050**	1.1284	1.1402
			15	**1.1690**	1.1840	1.2086
	Netflix Tiny	Mean	5	**0.8502**	0.8725	0.8915
			10	**0.8335**	0.8522	0.8621
			15	**0.8528**	0.8731	0.8927
		Min	5	**0.9888**	0.9991	1.0152
			10	**0.9855**	0.9974	1.0130
			15	**1.1857**	1.1965	1.2192

The case study comprises the evaluation in two well-known recommendation datasets:

- The MovieLens 100K dataset[1], which was collected by GroupLens Research Project at the University of Minnesota. It is composed of 100,000 ratings given by 943 users over 1682 movies in the five stars domain.
- The Netflix Tiny dataset, composed of 4427 users, 1000 movies, and 56,136 ratings, which were also given in the five stars domain. This is a smaller version of Netflix dataset, and it is available in the Personalised Recommendation Algorithms Toolkit[2].

4.2. Results

The results of the experimental procedure are presented in this section. First, we perform an optimisation of the parameters of NNMG-FT. After that, we analyse the results of the considered NNM approaches for recommendation aggregation GRSs and for rating aggregation GRSs. Finally, MAE improvement per group of NNMG-FT is analysed.

4.2.1. Parameter optimisation of the NNMG-FT

NNMG-FT has three parameters whose values need to be determined to adjust it. These parameters are k, δ_1 and δ_2, and their values are determined through experiments to determine the best value for each of them.

Parameter k ranges from 0 to 1 and it is used to determine how fuzzy profiles of users and items are modified to highlight rating tendencies. Table 5 shows the MAE of NNMG-FT in various group recommendation scenarios. The whole range of values for k was evaluated, here only values 0.35, 0.50 and 0.75 are shown for the sake of clearness. The results determine that NNMG-FT obtains the best MAE for $k = 0.35$.

Parameter δ_1 ranges from 0 to 2 (Yera et al., 2016), and the larger its value the more close a user and item profile have to be in order to consider them as matching tendencies. Table 6 shows the MAE of NNMG-FT in various configurations for dataset, aggregation approach, aggregation strategy and group size. The whole range of values for δ_1 was evaluated, here only values 0.9, 1.0 and 1.1 are shown for the sake of clearness. The results show that NNMG-FT obtains the best outcomes for $\delta_1 = 1.0$ in most of the evaluated scenarios.

Parameter δ_2 range from 0 to 2 (Yera et al., 2016), and the larger its value the more a rating has to deviate from its corresponding user-item tendency to be considered as noisy. Table 7 shows the MAE of NNMG-FT in various configurations. The whole range of values for δ_2 was evaluated, here only values 0.9, 1.0 and 1.1 are shown for the sake of clearness. The results show that NNMG-FT obtains the best outcomes for $\delta_2 = 1.0$ in most of the evaluated scenarios.

The parameter optimisation results determine that the best configuration for NNMG-FT parameters are $k = 0.35$, $\delta_1 = 1$ and $\delta_2 = 1$. The remaining experiments are performed with those values.

4.2.2. Noise management in recommendation aggregation GRSs

Table 8 shows the MAE of the recommendation aggregation GRSs with the compared NNM approaches. The lower the MAE of the GRS, the better the NNM approach. The best NNM of each configuration is highlighted in bold. NNMG-FT achieved the best results as compared with other approaches in all evaluated scenarios.

Beyond this general improvement, there were differences regarding the relative improvement across datasets, aggregation strategies and group sizes. Table 9 shows, for the various configurations of datasets, aggregation strategies and group sizes, the relative improvement of the NNM approaches. Additionally, for the comparison between NNM-Crisp and NNM-FT, it is shown both the relative improvement and the p-value of the Wilcoxon signed-rank test (significant values with $\alpha = 0.05$ are highlighted). The relative improvement has been calculated dividing the MAE of the first technique by the MAE of the reference technique. Wilcoxon test has been performed comparing the paired samples of each NNM approach.

In recommendation aggregation with average strategy, the relative improvement was uniform across group sizes (see Table 9).

[1] http://grouplens.org/datasets/movielens/100k/.
[2] http://prea.gatech.edu.

Table 6
NNMG-FT parameter optimisation. Optimisation of δ_1 using MAE.

Aggregation approach	Dataset	Aggregation strategy	Group size	δ_1		
				0.9	1.0	1.1
Rating aggregation	MovieLens 100k	Mean	5	0.8367	**0.8364**	0.8365
			10	0.8595	**0.8590**	0.8591
			15	0.8690	**0.8686**	0.8687
		Min	5	**0.8545**	**0.8545**	0.8547
			10	0.9179	**0.9177**	0.9183
			15	0.9808	**0.9805**	0.9810
	Netflix Tiny	Mean	5	0.8278	**0.8272**	0.8274
			10	0.8520	**0.8517**	**0.8517**
			15	0.8569	**0.8566**	0.8567
		Min	5	0.8558	0.8555	**0.8552**
			10	0.9172	0.9167	**0.9166**
			15	0.9599	0.9595	**0.9594**
Recomm. aggregation	MovieLens 100k	Mean	5	0.8340	**0.8335**	0.8340
			10	0.8595	**0.8590**	0.8591
			15	0.8804	**0.8802**	**0.8802**
		Min	5	0.9860	**0.9855**	0.9857
			10	1.1055	**1.1050**	1.1061
			15	1.1694	**1.1690**	1.1693
	Netflix Tiny	Mean	5	0.8504	**0.8502**	**0.8502**
			10	0.8340	**0.8335**	0.8340
			15	0.8532	**0.8528**	**0.8528**
		Min	5	**0.9885**	0.9888	0.9888
			10	0.9860	**0.9855**	0.9857
			15	1.1866	**1.1857**	1.1860

Table 7
NNMG-FT parameter optimisation. Optimisation of δ_2 using MAE.

Aggregation approach	Dataset	Aggregation strategy	Group size	δ_2		
				0.9	1.0	1.1
Rating aggregation	MovieLens 100k	Mean	5	0.8371	**0.8367**	**0.8367**
			10	0.8599	**0.8595**	0.8596
			15	0.8696	0.8690	**0.8689**
		Min	5	0.8549	**0.8545**	0.8546
			10	0.9183	**0.9179**	0.9181
			15	0.9813	**0.9808**	0.9809
	Netflix Tiny	Mean	5	0.8279	**0.8276**	0.8278
			10	0.8525	0.8520	**0.8519**
			15	0.8574	0.8569	**0.8567**
		Min	5	0.8559	0.8558	**0.8557**
			10	0.9176	**0.9171**	0.9172
			15	0.9603	**0.9599**	**0.9599**
Recomm. aggregation	MovieLens 100k	Mean	5	0.8507	0.8504	**0.8503**
			10	0.8725	**0.8721**	0.8722
			15	0.8806	0.8804	**0.8803**
		Min	5	0.9888	**0.9885**	**0.9885**
			10	1.1062	**1.1055**	1.1061
			15	1.1701	**1.1694**	1.1698
	Netflix Tiny	Mean	5	0.8343	**0.8340**	0.8341
			10	0.8343	**0.8340**	0.8341
			15	0.8535	0.8532	**0.8530**
		Min	5	0.9863	**0.9860**	**0.9860**
			10	0.8525	0.8520	**0.8519**
			15	1.1873	**1.1866**	1.1870

Moreover, almost all differences were statistically significant. Comparing the results across datasets, the relative improvement achieved in MovieLens dataset was larger than the improvement in Netflix Tiny (see Figs. 4a and 5a). This fact could be related to the lower sparsity of MovieLens, which results in a better characterisation of users and items through their fuzzy profiles.

In the results of the GRS with recommendation aggregation with least misery strategy, the improvement was larger than the improvement for average strategy (see Table 9 and Figs. 4b and 5b). This might be due to the higher sensitivity to noise of least misery strategy, e.g., the noise on a single rating can bias the recommendation. In contrast, in average strategy (see Figs. 4a and 5a) the improvement is lower because the impact of noisy ratings seems to be reduced by the aggregation approach itself.

In recommendation aggregation with least misery strategy, the improvement was larger for larger groups (see Table 9 and Figs. 4b and 5b). The increased improvement in this scenario was due to least misery being more sensitive to noisy values and large groups having higher chance of noisy ratings. Hence, NNMG-FT provides a NNM that benefits more the recommendation to larger groups with least misery strategy.

4.2.3. Noise management in rating aggregation GRSs

Table 10 shows the MAE of rating aggregation GRSs with the compared NNM approaches. The best NNM approach is highlighted

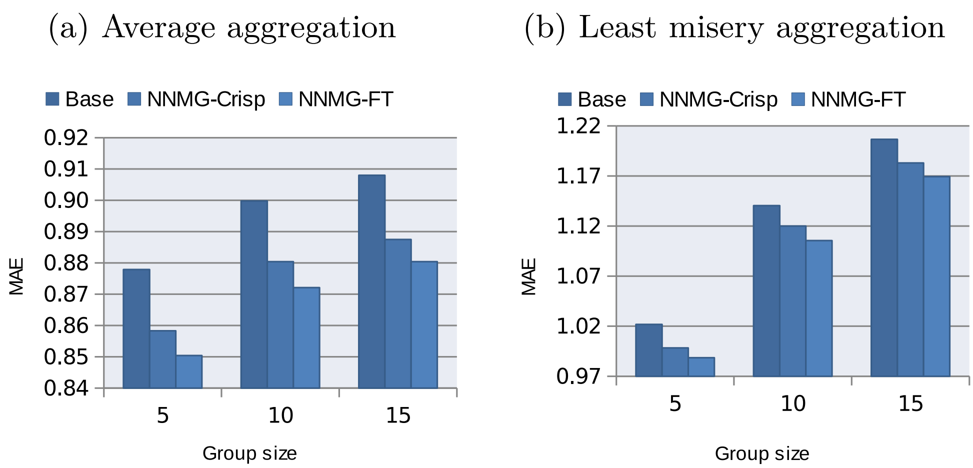


Fig. 4. Results of recommendation aggregation GRS on MovieLens 100k dataset.

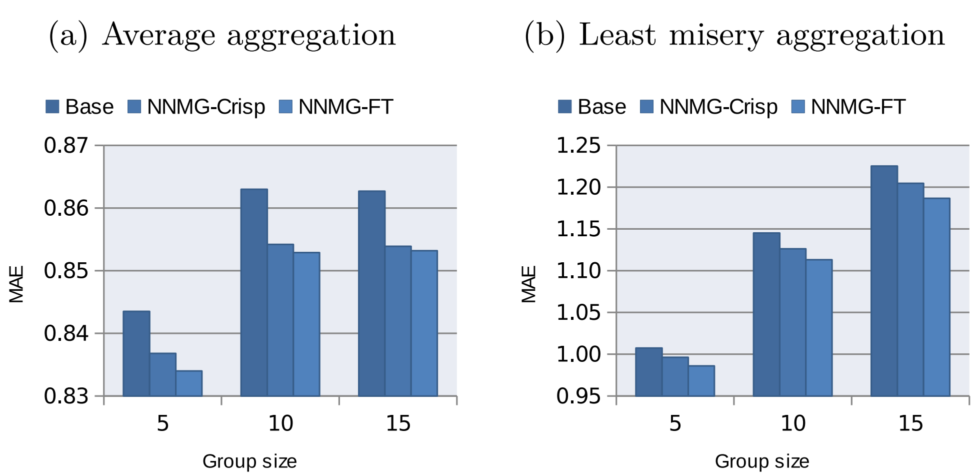


Fig. 5. Results of recommendation aggregation GRS on Netflix Tiny dataset.

Table 8
MAE of NNMG approaches on recommendation aggregation.

Dataset	Aggregation strategy	Group size	Base	NNMG-Crisp (Castro et al., 2017)	NNMG-FT
MovieLens 100k	Avg	5	0.8779	0.8583	**0.8504**
		10	0.8998	0.8804	**0.8721**
		15	0.9080	0.8875	**0.8804**
	Min	5	1.0218	0.9983	**0.9885**
		10	1.1404	1.1200	**1.1055**
		15	1.2066	1.1830	**1.1694**
Netflix Tiny	Avg	5	0.8435	0.8368	**0.8340**
		10	0.8630	0.8542	**0.8529**
		15	0.8627	0.8539	**0.8532**
	Min	5	1.0074	0.9963	**0.9860**
		10	1.1451	1.1262	**1.1131**
		15	1.2252	1.2046	**1.1866**

in bold. NNMG-FT obtained the best results in the majority of scenarios.

Beyond this general improvement, there were differences regarding the relative improvement across datasets, aggregation strategies and group sizes. Table 11 shows in the rows the configurations of the GRS evaluated, and in the columns, the relative improvement of the pairwise comparison of the NNM approaches. For the comparison between NNM-Crisp and NNM-FT, in addition to the relative improvement, it is shown the p-value of the Wilcoxon signed-rank test (significant values with $\alpha = 0.05$ are highlighted). The relative improvement has been calculated dividing the MAE of the first technique by the MAE of the reference technique. Wilcoxon test has been performed comparing the paired samples of each NNM approach.

In rating aggregation with average strategy (see Figs. 6a and 7a), the improvement was uniform across datasets and group sizes. Moreover, the majority of differences are statistically significant. The relative improvement of results were different to the

Table 9
Relative improvement of the pairwise comparison of NNMG approaches on recommendation aggregation. Note that, for the comparison of NNM-Crisp and NNM-FT, the p-value of Wilcoxon signed-rank test is shown (statistically significant values with $\alpha = 0.05$ are highlighted).

Dataset	Aggregation strategy	Group size	NNMG-Crisp vs Base	NNMG-FT vs Base	NNMG-Crisp vs NNMG-FT	
					Rel. imp.	*p*-value
MovieLens 100k	Avg	5	2.23%	3.13%	0.92%	**<.001**
		10	2.16%	3.08%	0.94%	**<.001**
		15	2.26%	3.04%	0.80%	**<.001**
	Min	5	2.30%	3.26%	0.98%	**<.001**
		10	1.79%	3.06%	1.29%	**<.001**
		15	1.96%	3.08%	1.15%	**<.001**
Netflix Tiny	Avg	5	0.79%	1.13%	0.33%	**<.001**
		10	1.02%	1.17%	0.15%	**.0499**
		15	1.02%	1.10%	0.08%	.0545
	Min	5	1.10%	2.12%	1.03%	**<.001**
		10	1.65%	2.79%	1.16%	**<.001**
		15	1.68%	3.15%	1.49%	**<.001**

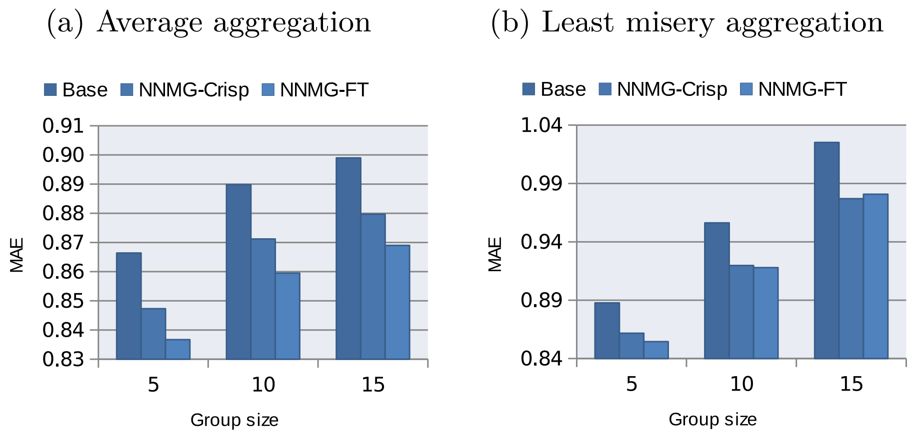


Fig. 6. Results of rating aggregation GRS on MovieLens 100k dataset.

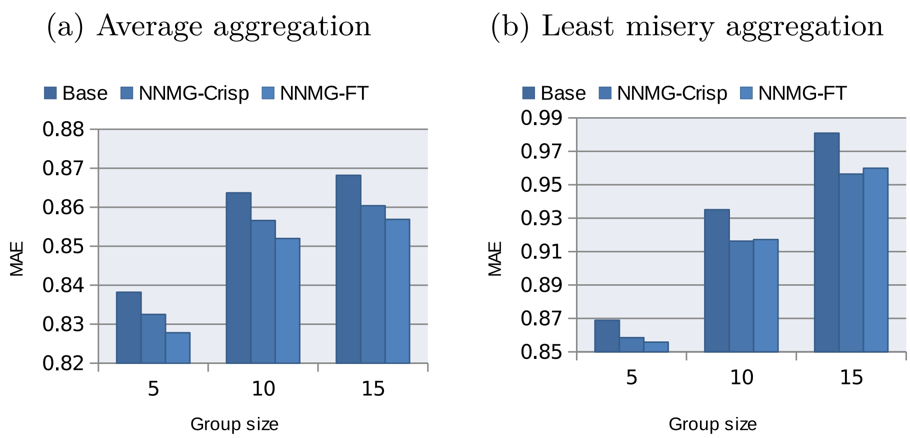


Fig. 7. Results of rating aggregation GRS on Netflix Tiny dataset.

Table 10
MAE of NNMG approaches on rating aggregation.

Dataset	Aggregation strategy	Group size	Base	NNMG-Crisp (Castro et al., 2017)	NNMG-FT
	Avg	5	0.8664	0.8473	**0.8367**
		10	0.8898	0.8712	**0.8595**
MovieLens		15	0.8990	0.8797	**0.8690**
100k	Min	5	0.8877	0.8617	**0.8545**
		10	0.9563	0.9197	**0.9179**
		15	1.0252	**0.9770**	0.9808
	Avg	5	0.8382	0.8325	**0.8278**
		10	0.8637	0.8566	**0.8520**
Netflix		15	0.8682	0.8604	**0.8569**
Tiny	Min	5	0.8689	0.8585	**0.8558**
		10	0.9351	**0.9163**	0.9172
		15	0.9809	**0.9564**	0.9599

Table 11
Relative improvement of the pairwise comparison of NNMG approaches on rating aggregation. Note that, for the comparison of NNM-Crisp and NNM-FT, the p-value of Wilcoxon signed-rank test is shown (statistically significant values with $\alpha = 0.05$ are highlighted).

Dataset	Aggregation strategy	Group size	NNMG-Crisp vs Base	NNMG-FT vs Base	NNMG-Crisp vs NNMG-FT	
					Rel. imp.	p-value
	Avg	5	2.20%	3.43%	1.25%	**<.001**
		10	2.09%	3.41%	1.34%	**<.001**
MovieLens		15	2.15%	3.34%	1.22%	**<.001**
100k	Min	5	2.93%	3.74%	0.84%	**<.001**
		10	3.83%	4.02%	0.20%	**.006**
		15	4.70%	4.33%	−0.39%	**<.001**
	Avg	5	0.68%	1.24%	0.56%	**<.001**
		10	0.82%	1.35%	0.54%	**<.001**
Netflix		15	0.90%	1.30%	0.41%	**<.001**
Tiny	Min	5	1.20%	1.51%	0.31%	**.016**
		10	2.01%	1.91%	−0.10%	.162
		15	2.50%	2.14%	−0.37%	**.001**

ones obtained for recommendation aggregation (see Figs. 4a and 5a), where NNMG-FT provided a higher relative improvement for larger groups. Here, there were no remarkable differences in the improvement among group sizes, therefore, all groups benefited from the application of NNMG-FT.

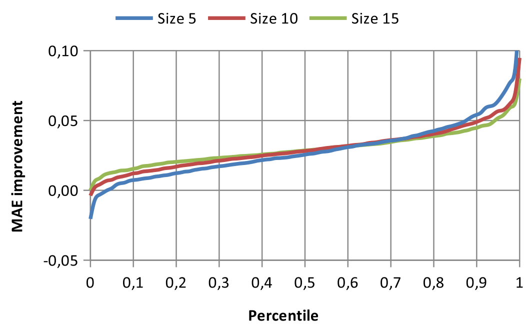
In rating aggregation with least misery strategy (see Figs. 6a and 7a), NNMG-FT achieved a higher relative improvement compared to the baseline. It also improved the recommendations in smaller group sizes as compared to NNMG-Crisp (see Fig. 6b and 7b).

Overall, the results show that NNMG-FT improved recommendations for both GRSs aggregation approaches. The improvements were greater in recommendation aggregation GRSs than in rating aggregation. This difference might be caused by the rating aggregation, which might implicitly remove some noise. The improvements achieved in both scenarios justify the application of NNMG-FT in both aggregation-based GRS approaches to improve recommendations targeted to groups removing natural noisy ratings.

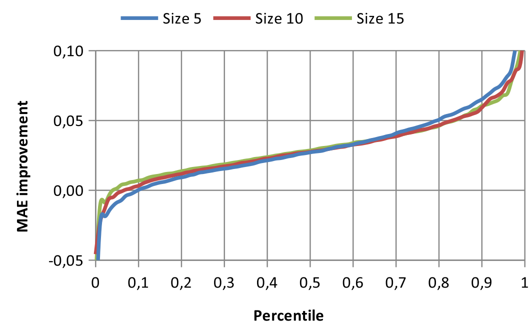
4.2.4. Analysis of MAE improvement per group

The application of the proposal in group recommendation improves the MAE overall. However, it is important to evaluate how the improvement is distributed among groups, this is, to

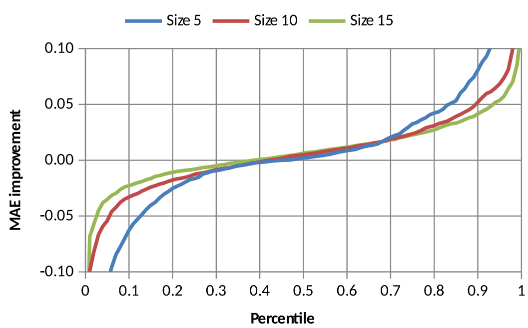
(a) Mean aggr. on MovieLens 100k.



(b) Min. aggr. on MovieLens 100k.



(c) Mean aggr. on Netflix Tiny.



(d) Min. aggr. on Netflix Tiny.

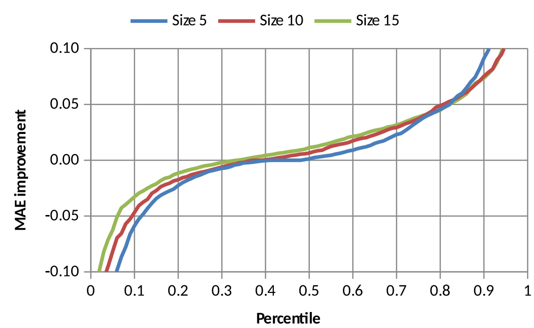
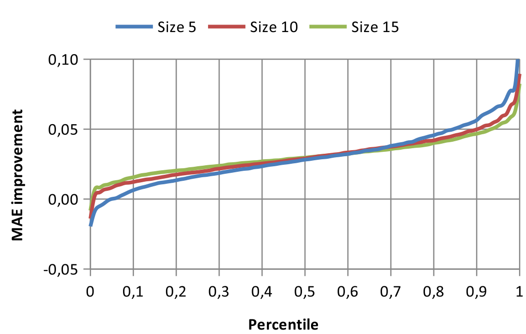
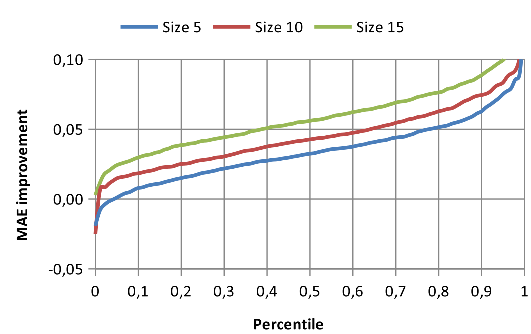


Fig. 8. MAE improvement per group shown as percentile for recommendation aggregation GRS.

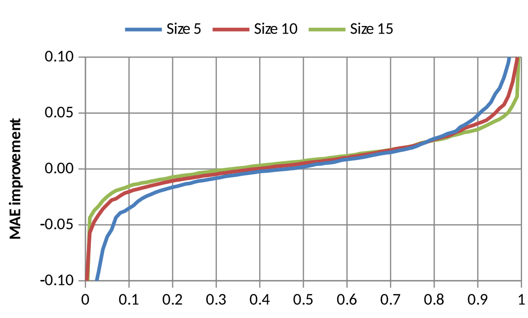
(a) Mean aggr. on MovieLens 100k.



(b) Min. aggr. on MovieLens 100k.



(c) Mean aggr. on Netflix Tiny.



(d) Min. aggr. on Netflix Tiny.

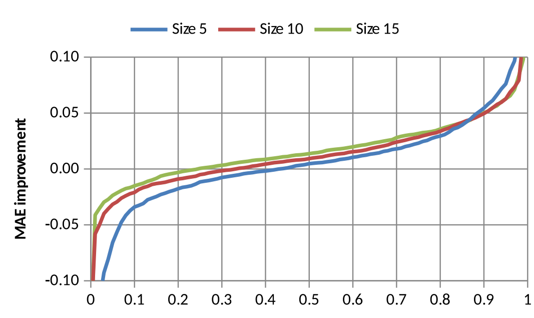


Fig. 9. MAE improvement per group shown as percentile for rating aggregation GRS.

evaluate whether there are groups whose recommendations are better while other groups receive worse recommendations. MAE improvement per group is analysed to provide a deeper study of the proposal.

Fig. 8 contains the results of MAE improvement per group for recommendation aggregation GRSs. Each diagram shows the results for each aggregation strategy and dataset, and they show the distribution of MAE improvement of NNMG-FT over Base. Fig. 8a and b show the results for MovieLens dataset, and it can be noticed that less than 5% of groups received worse recommendations with the proposal. Regarding Netflix Tiny dataset, shown in Fig. 8c and d, less than 21% of groups had a 0.02 decay in MAE.

Regarding rating aggregation GRSs, Fig. 9 contains the MAE improvement per group. Similarly, each diagram shows the results for each aggregation strategy and dataset, and they show the distribution of MAE improvement per group. Fig. 9a and b show the results for MovieLens dataset, where less than 10% of groups received worse recommendations with the proposal. Regarding Netflix Tiny dataset, shown in Fig. 9c and d, less than 18% of groups had a decay in MAE greater than 0.02.

Overall, results of MAE improvement per group show that, although there are some groups that receive worse recommendations with the proposal, the greater amount of groups receive better recommendations. Moreover, the magnitude of these improvements suggests that the proposal is suitable to be applied in group recommendation scenarios without a major decay in recommendation quality.

5. Conclusions

This paper proposes a natural noise management approach for group recommender systems using fuzzy tools (NNMG-FT). Specifically, NNMG-FT uses fuzzy profiles to characterise the rating tendency of users and items. With this characterisation, ratings that do not follow their corresponding user and item tendency are identified as noisy and, therefore, corrected. NNMG-FT performs two phases of noise correction: the first one follows a global approach, and the second is personalised to the target group.

A case study has been performed to compare NNMG-FT with previous natural noise management approaches. The results show that the management of natural noise with our proposal leads to improved results in the majority of evaluation scenarios, which comprise various aggregation approaches, aggregation strategies and group sizes. Moreover, a deeper study of the proposal showed that the improvement of recommendations is general and few groups had a decay in recommendation quality.

The study shows that NNMG-FT is beneficial for group recommendation. In order to further improve the NNM in future works, it is worth to study temporal dynamics, which enhance user preference modelling. Consideration of temporal dynamics would help at both detecting more noisy ratings and avoiding false positives, and therefore improve the detection of noise.

Future works will also focus on exploring NNM in context-aware scenarios. Context in recommender systems is characterised by its heterogeneity, covering very diverse information sources, such as temporal information, companion, or weather. Moreover, context-awareness leads to a higher sparsity of ratings. Therefore, specific researches are needed to study the particularities of context-aware scenario, in order to characterise natural noise in group recommender systems databases.

Acknowledgements

This paper was partially supported by the Spanish FPU fellowship (FPU13/01151), the Spanish National research project TIN2015-66524-P.

References

Adomavicius, G., & Tuzhilin, A. T. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering, 17*(6), 734–749.

Amatriain, X., Lathia, N., Pujol, J. M., Kwak, H., & Oliver, N. (2009a). The wisdom of the few: A collaborative filtering approach based on expert opinions from the web. In *32nd international ACM SIGIR conference* (pp. 532–539). New York, NY, USA: ACM.

Amatriain, X., Pujol, J. M., & Oliver, N. (2009b). I like it...i like it not: Evaluating user ratings noise in recommender systems. In *User modeling, adaptation, and personalization* (pp. 247–258). Springer.

Amatriain, X., Pujol, J. M., Tintarev, N., & Oliver, N. (2009c). Rate it again: Increasing recommendation accuracy by user re-rating. In *Third ACM conference on recommender systems* (pp. 173–180). ACM.

Ardissono, L., Goy, A., Petrone, G., Segnan, M., & Torasso, P. (2003). Intrigue: Personalized recommendation of tourist attractions for desktop and hand held devices. *Applied Artificial Intelligence, 17*(8–9), 687–714. doi:10.1080/713827254.

Bellogín, A., Said, A., & de Vries, A. P. (2014). The magic barrier of recommender systems–no magic, just ratings. In *International conference on user modeling, adaptation, and personalization* (pp. 25–36). Springer.

Castro, J., Quesada, F. J., Palomares, I., & Martínez, L. (2015). A consensus-driven group recommender system. *International Journal of Intelligent Systems, 30*(8), 887–906. doi:10.1002/int.21730.

Castro, J., Yera, R., & Martínez, L. (2017). An empirical study of natural noise management in group recommendation systems. *Decision Support Systems, 94*, 1–11.

Centeno, R., Hermoso, R., & Fasli, M. (2015). On the inaccuracy of numerical ratings: Dealing with biased opinions in social networks. *Information Systems Frontiers, 17*(4), 809–825.

Crossen, A., Budzik, J., & Hammond, K. J. (2002). Flytrap: Intelligent group music recommendation. In *Proceedings of the 7th international conference on intelligent user interfaces*. In *IUI '02* (pp. 184–185). New York, NY, USA: ACM. doi: 10.1145/502716.502748.

De Pessemier, T., Dooms, S., & Martens, L. (2014). Comparison of group recommendation algorithms. *Multimedia Tools and Applications, 72*(3), 2497–2541. doi:10.1007/s11042-013-1563-0.

Ekstrand, M. D., Riedl, J. T., & Konstan, J. A. (2011). Collaborative filtering recommender systems. *Foundations and Trends in Human-Computer Interaction, 4*(2), 81–173.

Garcia, I., Pajares, S., Sebastia, L., & Onaindia, E. (2012). Preference elicitation techniques for group recommender systems. *Information Sciences, 189*, 155–175.

Guo, F., & Dunson, D. B. (2015). Uncovering systematic bias in ratings across categories: A Bayesian approach. In *Proceedings of the 9th ACM conference on recommender systems* (pp. 317–320). ACM.

Herlocker, J., Konstan, J., Terveen, K., & Riedl, J. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems, 22*(1), 5–53.

Herrera, F., & Martinez, L. (2000). A 2-tuple fuzzy linguistic representation model for computing with words. *IEEE Transactions on Fuzzy Systems, 8*(6), 746–752. doi:10.1109/91.890332.

Kagita, V. R., Pujari, A. K., & Padmanabhan, V. (2015). Virtual user approach for group recommender systems using precedence relations. *Information Sciences, 294*, 15–30. Innovative Applications of Artificial Neural Networks in Engineering.

Kluver, D., Nguyen, T. T., Ekstrand, M., Sen, S., & Riedl, J. (2012). How many bits per rating? In *Proceedings of the sixth ACM conference on recommender systems* (pp. 99–106). ACM.

Koren, Y. (2010). Collaborative filtering with temporal dynamics. *Interacting with Computers of the ACM, 53*(4), 89–97.

Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer, 42*(8), 30–37.

Li, B., Chen, L., Zhu, X., & Zhang, C. (2013). Noisy but non-malicious user detection in social recommender systems. *World Wide Web, 16*(5–6), 677–699.

Martínez, L., Castro, J., & Yera, R. (2016). Managing natural noise in recommender systems. In C. Martín-Vide, T. Mizuki, & M. A. Vega-Rodríguez (Eds.), *Theory and practice of natural computing: 5th International conference, TPNC 2016, Sendai, Japan, December 12–13, 2016, proceedings* (pp. 3–17). Springer International Publishing. doi: 10.1007/978-3-319-49001-4_1.

Masthoff, J. (2015). Group recommender systems: Aggregation, satisfaction and group attributes. In F. Ricci, L. Rokach, & B. Shapira (Eds.), *Recommender systems handbook* (pp. 743–776). Springer US. doi: 10.1007/978-1-4899-7637-6_22.

McCarthy, J. F., & Anagnost, T. D. (1998). Musicfx: An arbiter of group preferences for computer supported collaborative workouts. In *The 1998 ACM conference on computer supported cooperative work* (pp. 363–372). ACM.

Ning, X., Desrosiers, C., & Karypis, G. (2015). A comprehensive survey of neighborhood-based recommendation methods. In F. Ricci, L. Rokach, & B. Shapira (Eds.), *Recommender systems handbook* (pp. 37–76). Springer US. doi: 10.1007/978-1-4899-7637-6_2.

O'Connor, M., Cosley, D., Konstan, J. A., & Riedl, J. (2001). Polylens: A recommender system for groups of users. In *Proceeding of the 7th conference on European conference on computer supported cooperative work (ECSCW'01)* (pp. 199–218). Bonn, Germany – September 16–20.

O'Mahony, M. P., Hurley, N. J., & Silvestre, G. (2006). Detecting noise in recommender system databases. In *11th international conference on intelligent user interfaces* (pp. 109–115). ACM.

Ortega, F., Hernando, A., Bobadilla, J., & Kang, J. H. (2016). Recommending items to group of users using matrix factorization based collaborative filtering. *Information Sciences, 345*, 313–324.

Pham, H. X., & Jung, J. J. (2013). Preference-based user rating correction process for interactive recommendation systems. *Multimedia Tools and applications, 65*(1), 119–132.

Rafailidis, D., Kefalas, P., & Manolopoulos, Y. (2017). Preference dynamics with multimodal user-item interactions in social media recommendation. *Expert Systems with Applications, 74*, 11–18. https://doi.org/10.1016/j.eswa.2017.01.005.

Rodríguez, R. M., Labella, A., & Martínez, L. (2016). An overview on fuzzy modelling of complex linguistic preferences in decision making. *International Journal of Computational Intelligence Systems, 9*(sup1), 81–94. doi:10.1080/18756891.2016.1180821.

Rodríguez, R. M., & Martínez, L. (2013). An analysis of symbolic linguistic computing models in decision making. *International Journal of General Systems, 42*(1), 121–136. doi:10.1080/03081079.2012.710442.

Saia, R., Boratto, L., & Carta, S. (2016). A semantic approach to remove incoherent items from a user profile and improve the accuracy of a recommender system. *Journal of Intelligent Information Systems, 47*(1), 111–134.

Said, A., Berkovsky, S., & De Luca, E. W. (2011). Group recommendation in context. In *2nd Challenge on context-aware movie recommendation*. In *CAMRa '11* (pp. 2–4). New York, NY, USA: ACM.

Said, A., Jain, B. J., Narr, S., & Plumbaum, T. (2012). Users and noise: The magic barrier of recommender systems. In J. Masthoff, B. Mobasher, M. C. Desmarais, & R. Nkambou (Eds.), *User modeling, adaptation, and personalization: 20th International conference, UMAP 2012, Montreal, Canada, July 16–20, 2012. Proceedings* (pp. 237–248)). Springer Berlin Heidelberg. doi: 10.1007/978-3-642-31454-4_20.

Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. In *10th International conference on world wide web* (pp. 285–295). ACM.

Schweizer, B., & Sklar, A. (2011). *Probabilistic metric spaces*. Courier Corporation.

Trang Tran, T. N., Atas, M., Felfernig, A., & Stettinger, M. (2017). An overview of recommender systems in the healthy food domain. *Journal of Intelligent Information Systems*. doi:10.1007/s10844-017-0469-0.

Vaca, C. K., Mantrach, A., Jaimes, A., & Saerens, M. (2014). A time-based collective factorization for topic discovery and monitoring in news. In *Proceedings of the 23rd international conference on world wide web*. In *WWW '14* (pp. 527–538). New York, NY, USA: ACM. doi: 10.1145/2566486.2568041.

Yager, R. R., & Zadeh, L. A. (2012). *An introduction to fuzzy logic applications in intelligent systems*: Vol. 165. Springer Science & Business Media.

Yera, R., Castro, J., & Martínez, L. (2016). A fuzzy model for managing natural noise in recommender systems. *Applied Soft Computing, 40*, 187–198.

Yera, R., & Martínez, L. (2017). Fuzzy tools in recommender systems: A survey. *International Journal of Computational Intelligence Systems, 10*(1), 776–803.

Yera Toledo, R., Caballero Mota, Y., & Martínez, L. (2015). Correcting noisy ratings in collaborative recommender systems. *Knowledge-Based Systems, 76*, 96–108.

Yu, P., Lin, L., & Yao, Y. (2016). A novel framework to process the quantity and quality of user behavior data in recommender systems. In B. Cui, N. Zhang, J. Xu, X. Lian, & D. Liu (Eds.), *Web-age information management: 17th International conference, WAIM 2016, Nanchang, China, June 3–5, 2016, Proceedings, Part I* (pp. 231–243)). Springer International Publishing. doi: 10.1007/978-3-319-39937-9_18.

Yu, Z., Zhou, X., Hao, Y., & Gu, J. (2006). Tv program recommendation for multiple viewers based on user profile merging. *User Modeling and User-Adapted Interaction, 16*(1), 63–82. doi:10.1007/s11257-006-9005-6.

Zadeh, L. A. (1965). Fuzzy sets. *Information and control, 8*(3), 338–353.

Zadeh, L. A. (2012). Computing with words: Principal concepts and ideas. *Studies in Fuzziness and Soft Computing*. Springer.

Zhang, C., Wang, K., Yu, H., Sun, J., & Lim, E.-P. (2014). Latent factor transition for dynamic collaborative filtering. In *Proceedings of the 2014 siam international conference on data mining* (pp. 452–460). SIAM. doi: 10.1137/1.9781611973440.52.

Zhang, X., Zhao, J., & Lui, J. (2017). Modeling the assimilation-contrast effects in online product rating systems: Debiasing and recommendations. In *Proceedings of the eleventh ACM conference on recommender systems* (pp. 98–106). ACM.

Zimmermann, H.-J. (2001). *Fuzzy set theory and its applications*. Springer Science & Business Media.